GOVERNMENT OF NAGALAND
DIRECTORATE OF TECHNICAL EDUCATION
NAGALAND: KOHIMA

DIPLOMA CURRICULUM STRUCTURE FOR COMPUTER SCIENCE AND ENGINEERING (CSE) 3 (THREE) YEARS COURSE

APPROVED
BY
STATE COUNCIL FOR TECHNICAL EDUCATION (SCTE)
NAGALAND

DESIGNED IN COLLABORATION WITH

- STRUCTURE BY NITTTR, KOLKATA
- AICTE MODEL CURRICULUM
- BASED ON NEP2020
- COA GUIDELINES AND NORMS

FOR THE STATE OF NAGALAND
(*July 2025*)

SAMPLE PATH FOR 3-YEAR DIPLOMA PROGRAMME IN COMPUTER SCIENCE AND ENGINEERING
SEMESTER-I

Sl. No	Category of Course	Code	Course Title	Study Scheme				Evaluation Scheme							Total Marks	Credit
				Pre-requisite	Contact Hours/ week			Theory				Practical				
								End Exam	Progressive Assessment			End Exam	Progressive Assessment			
					L	T	P		Mid Term Exam	Assignment	Attendance		Sessional	Viva voce		
1	Basic Science	BS101	Mathematics-I		2	1	0	60	20	15	5				100	3
2		BS103	Applied Physics-I		2	1	0	60	20	15	5				100	3
3		BS105	Applied Chemistry		2	1	0	60	20	15	5				100	3
4		BS107	Applied Physics-I Lab		0	0	2					40	40	20	100	1
5		BS109	Applied Chemistry Lab		0	0	2					40	40	20	100	1
6	Humanities & Social Science	HS101	Communication Skills in English-I		2	0	0	60	20	15	5				100	2
7		HS103	Sports and Yoga		0	0	2					40	40	20	100	1
8		HS105	Communication Skills in English-I Lab		0	0	2					40	40	20	100	1
9	Engineering Science	ES101	Engineering Graphics		0	0	3					40	40	20	100	1.5
10		ES103	Engineering Workshop Practice		0	0	3					40	40	20	100	1.5
					8	**3**	**14**								**1000**	**18**

SAMPLE PATH FOR 3-YEAR DIPLOMA PROGRAMME IN COMPUTER SCIENCE AND ENGINEERING
SEMESTER-II

Sl. No	Category of Course	Code	Course Title	Study Scheme				Evaluation Scheme							Total Marks	Credit
				Pre-requisite	Contact Hours/ week			Theory				Practical				
								End Exam	Progressive Assessment			End Exam	Progressive Assessment			
					L	T	P		Mid Term Exam	Assignment	Attendance		Sessional	Viva voce		
1	Basic Science	BS102	Mathematics-II	BS101	3	1	0	60	20	15	5				100	4
2		BS104	Applied Physics-II	BS103	2	1	0	60	20	15	5				100	3
3		BS106	Applied Physics-II Lab	BS107	0	0	2					40	40	20	100	1
4	Engineering Science	ES102	Introduction to IT Systems		2	0	0	60	20	15	5				100	2
5		ES104	Fundamentals of Electrical & Electronics Engineering		2	1	0	60	20	15	5				100	3
6		ES106	Engineering Mechanics		2	1	0	60	20	15	5				100	3
7		ES108	Introduction to IT Systems Lab		0	0	2					40	40	20	100	1
8		ES110	Fundamentals of Electrical & Electronics Engineering Lab		0	0	2					40	40	20	100	1
9		ES112	Engineering Mechanics Lab		0	0	2					40	40	20	100	1
10	Humanities & Social Science	HS106	Communication Skills in English-II Lab		0	0	2					-	-	-	-	1
11	Audit	AU102	Environmental Science		2	0	0									0
					13	**4**	**10**								**900**	**20**

SAMPLE PATH FOR 3-YEAR DIPLOMA PROGRAMME IN COMPUTER SCIENCE AND ENGINEERING
SEMESTER-III

SI. No	Category of Course	Code	Course Title	Study Scheme				Evaluation Scheme							Total Marks	Credit
				Pre-requisite	Contact Hours/ week			Theory				Practical				
								End Exam	Progressive Assessment			End Exam	Progressive Assessment			
					L	T	P		Mid Term Exam	Assignment	Attendance		Sessional	Viva voce		
1	Program core course	COPC201	Computer Programming	ES102	2	0	0	60	20	15	5				100	2
2		COPC203	Python Programming		2	0	0	60	20	15	5				100	2
3		COPC205	Data Structures		2	0	0	60	20	15	5				100	2
4		COPC207	Computer System Organization		3	1	0	60	20	15	5				100	4
5		COPC209	Algorithms		3	1	0	60	20	15	5				100	4
6		COPC211	Computer Programming Lab		0	0	4					40	40	20	100	2
7		COPC213	Python Programming Lab		0	0	4					40	40	20	100	2
8		COPC215	Data Structures Lab		0	0	2					40	40	20	100	1
9	Summer Internship-1 (2 weeks) after II-Sem	SI201	Summer Internship-1		0	0	0					40	40	20	100	1
					12	**2**	**10**								**900**	**20**

SAMPLE PATH FOR 3-YEAR DIPLOMA PROGRAMME IN COMPUTER SCIENCE AND ENGINEERING
SEMESTER-IV

Sl. No	Category of Course	Code	Course Title	Study Scheme				Evaluation Scheme							Total Marks	Credit
				Pre-requisite	Contact Hours/ week			Theory				Practical				
								End Exam	Progressive Assessment			End Exam	Progressive Assessment			
					L	T	P		Mid Term Exam	Assign ment	Atten dance		Sessio nal	Viva voce		
1	Program core course	COPC202	Operating Systems		3	0	0	60	20	15	5				100	3
2		COPC204	Introduction to DBMS	COPC205	3	0	0	60	20	15	5				100	3
3		COPC206	Computer Networks	ES102	3	0	0	60	20	15	5				100	3
4		COPC208	Software Engineering		3	0	0	60	20	15	5				100	3
5		COPC210	Web Technologies		3	0	0	60	20	15	5				100	3
6		COPC212	Operating Systems Lab		0	0	2					40	40	20	100	1
7		COPC214	Introduction to DBMS Lab	COPC211	0	0	2					40	40	20	100	1
8		COPC216	Computer Networks Lab		0	0	2					40	40	20	100	1
9		COPC218	Web Technologies Lab		0	0	2					40	40	20	100	1
10	Minor Project	PR202	Minor Project		0	0	4					40	40	10	100	2
11	Mandatory Course	AU202	Essence of Indian Knowledge and Tradition		2	0	0								0	0
					17	**0**	**12**								**1000**	**21**

SAMPLE PATH FOR 3-YEAR DIPLOMA PROGRAMME IN COMPUTER SCIENCE AND ENGINEERING
SEMESTER-V

Sl. No	Category of Course	Code	Course Title	Study Scheme				Evaluation Scheme							Total Marks	Credit
				Pre-requisite	Contact Hours/ week			Theory				Practical				
								End Exam	Progressive Assessment			End Exam	Progressive Assessment			
					L	T	P		Mid Term Exam	Assignment	Attendance		Sessional	Viva voce		
1	Program core course	COPC301	Introduction to e-Governance		2	1	0	60	20	15	5				100	3
2		COPC303	Internet of Things		2	1	0	60	20	15	5				100	3
3	Program Elective course	COPE301	Program Elective-1 (Mobile Computing)		3	0	0	60	20	15	5				100	3
4		COPE303	Program Elective-2 (Information Security)		3	0	0	60	20	15	5				100	3
5		COPE305	Program Elective-1 (Mobile Computing Lab)		0	0	2					40	40	20	100	1
6		COPE307	Program Elective-2 (Information Security Lab)		0	0	2					40	40	20	100	1
7	Open Elective	OE301	Open Elective-1 (Disaster Management)		3	0	0	60	20	15	5				100	3
8	Summer Internship-II (3 weeks) after IVth Sem	SI301	Summer Internship-2		0	0	0					40	40	20	100	4
9	Major Project	PR302		PR202	0	0	2									^
					13	2	6								**800**	**21**

SAMPLE PATH FOR 3-YEAR DIPLOMA PROGRAMME IN COMPUTER SCIENCE AND ENGINEERING
SEMESTER-VI

Sl. No	Category of Course	Code	Course Title	Study Scheme				Evaluation Scheme							Total Marks	Credit
				Pre-requisite	Contact Hours/ week			Theory				Practical				
								End Exam	Progressive Assessment			End Exam	Progressive Assessment			
					L	T	P		Mid Term Exam	Assignment	Attendance		Sessional	Viva voce		
1	Program Elective course	COPE302	Program Elective-3 (Multimedia Technologies)		3	0	0	60	20	15	5				100	3
2		COPE304	Program Elective-4 (FOSS)		3	0	0	60	20	15	5				100	3
3		COPE306	Program Elective-3 (Multimedia Technologies Lab)		0	0	2					40	40	20	100	1
4		COPE308	Program Elective-4 (FOSS Lab)		0	0	2					40	40	20	100	1
5	Humanities and Social Science course	HS302	Entrepreneurship and Start-ups		3	1	0	60	20	15	5				100	4
6	Open Elective	COOE302	Open Elective-2 (Artificial Intelligence)		3	0	0	60	20	15	5				100	3
7	Mandatory Course	AU302	Indian Constitution		2	0	0								0	0
8	Major Project	PR302	Project	PR302	0	0	6					100	50	50	200	4^
9	Seminar	SE302	Technical Seminar		1	0	0						50	50	100	1
					15	1	10								**900**	**20**

^1 credit is carried forward from the Vth semester major project evaluation.

Total Marks=5500 **Total Credits=120**

Semester - II

Introduction to IT Systems								
Course code: ES102				**Semester: II**				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Class Test**	**Assignment**	**Attendance**	
2	0	0	2	60	20	15	5	100
Rationale / Course Learning Objectives: -								
This course is intended to make new students comfortable with computing environment - Learning basic computer skills, learning basic application software tools, Understanding Computer Hardware, Cyber security awareness								

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	**Basic Internet skills**: Understanding browser, efficient use of search engines, awareness about Digital India portals (state and national portals) and college portals. General understanding of various computer hardware components – CPU, Memory, Display, Keyboard, Mouse, HDD, SSD and other Peripheral Devices.	6	11
2	**OS Installation (Linux and MS Windows)** Unix Shell: Introduction, vi editor Basic Unix Commands: ls, cp, mkdir, rmdir, cd, chmod, pwd	6	11
3	**HTML4 and CSS** HTML4: Introductions, Elements, Attributes, Headings, Paragraph, Styles, Formatting, Page Title, Links, Tables, List, Div, Classes, Forms and Media. CSS: Introductions, Syntax, Selectors, The Three Ways to Insert CSS, Colors, Background, Borders, Margins, Paddings, Outline, Text, Font, Icon, Link, Lists, Tables, Position, Z-Index, Float, Align. Develop basic personal webpage using HTML4 and CSS.	10	20
4	**Office Tools**: OpenOffice Writer, OpenOffice Spreadsheet (Calc), OpenOffice Impress.	5	9
5	**Information Security Basics** Introduction to Information Security, The CIA Triad, Best Practices of Information Security.	5	9
	TOTAL	**32**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	Basic Internet skills:	6	4	4	3	11
2	OS Installation (Linux and MS Windows)	6	3	4	4	11
3	HTML4 and CSS	10	5	5	10	20
4	Office Tools	5	3	3	3	9
5	Information Security Basics	5	4	3	2	9
	TOTAL	**32**	**19**	**19**	**22**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. Russell, Norvig, Artificial intelligence: A modern approach, 2nd edition. Pearson/Prentice Hall.
2. M.C. Trivedi, A classical approach to Artificial Intelligence, Khanna Publishing House, New Delhi (2018)
3. V.K. Jain, Machine Learning, Khanna Publishing House, New Delhi (2018)
4. Ethem Alpaydin, Introduction to Machine Learning, Second Edition, http://mitpress.mit.edu/catalog/item/default.asp?ttype=2&tid=12012.

Course Outcomes:

At the end of the course, the student will be able to:

CO1	Identify problems that are amenable to solution by AI methods.
CO2	Design and carry out an empirical evaluation of different algorithms on a problem formalization and state the conclusions that the evaluation supports.
CO3	Have a good understanding of the fundamental issues and challenges of machine learning: data, model selection, model complexity, etc.
CO4	Able to design and implement various machine learning algorithms in a range of real-world applications.
CO5	Appreciate the underlying mathematical relationships within and across Machine Learning algorithms and the paradigms of supervised and un-supervised learning.

Introduction to IT Systems Lab							
Course code: ES108				**Semester: II**			
Teaching Scheme				**Evaluation Scheme**			
Contact Hours/Week				**Theory**			
				End Term	**Progressive Assessment**		**Total Marks**
L	**T**	**P**	**Credit**		**Sessional**	**Viva Voce**	
0	0	2	1	40	40	20	100
Rationale / Course Learning Objectives: -							
This Lab course is intended to practice whatever is taught in theory class of 'Introduction of IT Systems and become proficient in using computing environment - basic computer skills, basic application software tools, Computer Hardware, cyber security features, etc.							

Course Content:

SL. NO	UNIT	HOURS
1	Browser features, browsing, using various search engines, writing search queries	4
2	Visit various e-governance/Digital India portals, understand their features, services offered	4
3	Read Wikipedia pages on computer hardware components, look at those components in lab, identify them, recognize various ports/interfaces and related cables, etc.	4
4	Install Linux and Windows operating system on identified lab machines, explore various options, do it multiple times	4
5	Connect various peripherals (printer, scanner, etc.) to computer, explore various features of peripheral and their device driver software.	4
6	Practice HTML commands, try them with various values, make your own Webpage	4
7	Explore features of Open Office tools, create documents using these features, do it multiple times	4
8	Explore security features of Operating Systems and Tools, try using them and see what happens.	4
	TOTAL	**32**

References:

1. Online resources, Linux man pages, Wikipedia.
2. R.S. Salaria, Computer Fundamentals, Khanna Publishing House.
3. Ramesh Bangia, PC Software Made Easy – The PC Course Kit, Khanna Publishing House.

4. Mastering Linux Shell Scripting: A practical guide to Linux command-line, Bash scripting, and Shell programming, by Mokhtar Ebrahim, Andrew Mallett.
5. IT Essentials PC Hardware and Software Companion Guide, Davis Anfinson and Ken Quamme, CISC Press, Pearson Education.
6. PC Hardware and A+ Handbook, Kate J. Chase PHI (Microsoft).

Course outcomes:

At the end of the course student will be able to comfortably work on computer, install and configure
OS, assemble a PC and connect it to external devices, write documents, create worksheets, prepare presentations, protect information and computers from basic abuses/attacks.

Semester - III

Computer Programming								
Course code: COPC201				**Semester:** III				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment (PA)**			**Total Marks**
L	**T**	**P**	**Credit**		**Mid-Term Exam**	**Assignment**	**Attendance**	
2	0	0	2	60	20	15	5	100

Rationale / Course Learning Objectives: -

To enable student, develop structured solutions to problems and implementing them using computers. This involves two parts: i) Formulating a solution for a given problem as a well-defined sequence of actions, and ii) Expressing solution in a machine-readable form or a programming language. For the second part, we will learn the common units of programming languages. The first part can only be learned through the repeated practice of solving problems

Course Content:

The language of choice will be C. The focus will be on problem solving and problem where these ideas can be applied. The main focus of the class will to take examples of problems where these ideas can be employed.

UNIT NO	UNIT	HOURS	MARKS
1	**INTRODUCTION**: Introduction to C, C Fundamentals: The 'C' Character Set identifiers, Keywords, Data Types, Variables and its Declaration.	4	6
2	**OPERATORS AND INPUT/OUTPUT STATEMENTS:** Arithmetic, Relational, Logical and Bitwise Operators, Formatted; Input, Output statements and File I/O; Create, Open, Write /Read and Close a File.	6	10
3	**CONTROL STATEMENT**: Conditional Statements; if statement, if-else statement, if-else if-else statement, nested if statement and switch statement, Repeat Statements; Loops – for loop, while loop and do-while loop.	8	16
4	**ARRAYS, STRINGS & FUNCTIONS**: Arrays, array declaration, Array and initialization, Multidimensional Arrays, Strings; string function (concatenation, comparison and length of a string) , Functions ; Function Prototypes, Passing Arguments to a Function and Recursion function .	8	18
5	**POINTERS**: Concepts of pointers, Pointer Declarations, Passing Pointers to Functions, Dynamic Memory Allocation.	6	10
	TOTAL	**32**	**60**

Specification Table Showing Distribution of Marks and Hours:

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	Introduction:	4	2	2	2	6
2	Operators and input/output statements:	6	4	4	2	10
3	Control statement:	8	4	8	4	16
4	Arrays, strings & functions	8	4	10	4	18
5	Pointers:	6	2	6	2	10
	TOTAL	**32**				**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. Let Us C, Yashavant Kanetkar
2. Problem Solving and Programming in C, R.S. Salaria, Khanna Publishing House
3. C Programming Absolute Beginner's Guide, Dean Miller and Greg Perry
4. The C Programming Language, Kernighan and Ritchie, Prentice Hall of India
5. Programming in ANSI C, E. Balagurusamy, Tata McGraw-Hill
6. C Programming & Data Structures, B. A. Fouruzan and R. F. Gilberg, CENGAGE Learning.
7. Outline of Programming with C, Byron Gottfried, Schaum, McGraw

Course Outcomes:

Student should be:

1. Able to identify C language fundamentals like keywords, identifiers, data types, operators, data input/output statements.
2. Able to apply conditional branching and looping statements.
3. Able to make use of arrays, strings, functions and pointers in C to solve problems.
4. Able to computationally formulate basic problems using C language.

Python Programming								
Course code: COPC203				**Semester: III**				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Class Test**	**Assignment**	**Attendance**	
2	0	0	2	60	20	15	5	100

Rationale / Course Learning Objectives: -

The proposed course aims to equip students with the foundational knowledge and practical skills necessary to effectively program using Python. By delving into the core concepts of programming, including variables, data types, control structures, functions, modules, file I/O, text processing, and regular expressions, students will be able to develop a solid understanding of the programming process.

Moreover, the course will explore the use of popular frameworks like Web2Py, Django, Ruby on Rails, or Struts, providing students with the tools and techniques to build robust and scalable applications. This practical component will enhance their ability to apply their programming knowledge to real-world projects.

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	**Introduction, Variables and Data Types** History, Features, Setting up path, Installation and Working with Perl/Python, Basic Syntax; Understanding Perl/Python variables, Numeric data types, Using string data type and string operations, Basic Operators, Understanding coding blocks, Defining list andlist slicing, Other Data Types(Tuples, List, Dictionary -Python, Arrays, Associative Arrays/Hashes - Perl)	3	5
2	**Control Structures** Conditional blocks using if, else and elseif, For loops and iterations, while loops, Loop manipulation using continue, break and else (and pass in Python), Programming using conditional and loops block	5	10
3	**Functions, Modules and Packages** Organizing Perl codes using functions, Organizing Perl projects intomodules, Importing own moduleas well as external modules, Understanding Packages	8	15
4	**File I/O, Text Processing, Regular Expressions** Understanding read functions, Understanding write functions, Programming using file operations, Powerful pattern matching	8	15

	and searching, Power of pattern searching using regex		
5	**Frameworks** Frameworks - Web2Py, Django, Ruby on Rails, Struts (any one ofthese or any other)	8	15
	TOTAL	**32**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	**C**	**A**	**Total Mark**
1	Introduction, Variables and Data Types	3	3	2	0	5
2	Control Structures	5	2	3	5	10
3	Functions, Modules and Packages	8	2	5	8	15
4	File I/O, Text Processing, Regular Expressions	8	5	6	4	15
5	Frameworks	8	2	5	8	15
	TOTAL	**32**	**14**	**21**	**25**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. Taming Python by Programming, Jeeva Jose, Khanna Publishing House
2. Starting Out with Python, Tony Gaddis, Pearson
3. Core Python Programming, Wesley J. Chun, Prentice Hall
4. Python Programming: Using Problem Solving Approach, Reema Thareja, Oxford University
5. Introduction to Computation and Programming Using Python. John V. Guttag, MIT Press.
6. Beginning Python using Python 2.6 and Python 3, James Payne, Wrox publishing
7. Practical Programming: An Introduction to Computer Science using Python 3, Paul Gries, The Pragmatic Bookshelf

Course Outcomes:

At the end of the course student will be able to build program with a scripting language and will be able to learn any other scripting language on their own.

Data Structures								
Course code: COPC205				**Semester:** III				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Mid-Term Exam**	**Assignment**	**Attendance**	
2	0	0	2	60	20	15	5	100

Rationale / Course Learning Objectives: -

The subject *Data Structures* is fundamental in computer science, enabling efficient data organization, storage, and manipulation. It provides a foundation for problem-solving by introducing structures like stacks, queues, linked lists, trees, and graphs. These structures optimize operations such as searching, sorting, and traversal, enhancing algorithm performance. Understanding data structures is crucial for software development, system design, and real-world applications like databases and networking. Mastery of these concepts equips students with essential skills for efficient programming and computational thinking.

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	**Introduction to Data Structures**: Basic Terminology, Classification of Data Structures, Operations on Data Structures.	3	5
2	**Linear Data Structures**- Stacks: Introduction to Stacks, Array Representation of Stacks, Operations ona Stack, Applications of Stacks-Infix-to-Postfix Transformation, evaluating Postfix Expressions. Queues: Introduction to Queues, Array Representation of Queues, Operations on a Queue, Types of Queues-DeQueue, Circular Queue, Applications of Queues-Round Robin Algorithm.	9	18
3	**Linked Lists**: Singly Linked List, Representation in Memory, Operations on a Single Linked List, Circular Linked Lists, Doubly Linked Lists, Linked List Representation and Operations of Stack, Linked List Representation and Operations of Queue.	8	17
4	**Non-Linear Data Structures** - Trees: Basic Terminologies, Definition and Concepts of Binary Trees, Representations of a Binary Tree using Arrays and Linked Lists, Operations on a Binary Tree-Insertion, Deletion, Traversals, Types of Binary Trees. GRAPHS: Graph Terminologies, Representation of Graphs- Set, Linked, Matrix, Graph Traversals	12	20
	TOTAL	**32**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	Introduction to Data Structures	3	3	2	0	**5**
2	Linear Data Structures	9	3	7	8	**18**
3	Linked Lists:	8	3	7	7	**17**
4	Non-Linear Data Structures	12	3	7	10	**20**
	TOTAL	**32**	**12**	**23**	**25**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. Data Structures, R.S. Salaria, Khanna Book Publishing, New Delhi
2. Data Structures Using C, Reema Thareja, Oxford University Press India.
3. Classic Data Structures, Samanta Debasis, Prentice Hall of India.
4. Fundamentals of Data Structure in C, Horowitz, Ellis, Sahni, Sartaj, Anderson-Freed, Susan, University Press, India.
5. Data Structures: A Pseudo code approach with C, Richard F. Gilberg, Behrouz A. Forouzan, CENGAGE Learning, India.
6. .Data Structures and Algorithms: Concepts, Techniques and Applications, G. A. V. Pai, McGraw- Hill Education, India.

Course Outcomes:

Have a good understanding of Data Structures and its applications in algorithms.

Computer System Organisation								
Course code: COPC207				**Semester: III**				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Mid-Term Exam**	**Assignment**	**Attendance**	
3	1	0	4	60	20	15	5	100
Rationale / Course Learning Objectives: -								
To have a thorough understanding of the basic structure and operation of a digital computer, its architectures, and computational designs.								

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	Structure of Computers: Computer Functional units, Von-Neumann architecture, Bus structures, Basic Operational Concepts, Error detecting codes. Register Transfer and Micro Operations: Register transfer, Bus and memory transfers, Arithmetic micro-operations, Logic micro-operations, Shift micro-operations, and Arithmetic logic shift unit.	13	12
2	Micro Programmed Control: Control memory, Address sequencing, and design of control unit. Computer Arithmetic: Addition and Subtraction, Multiplication and Division algorithms, Floating-point arithmetic operation, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline Vector Processing, Array Processors.	12	11
3	Introduction to Microprocessor Architecture: Instruction Set Architecture design principles from programmer's perspective. One example microprocessor (Intel, ARM, etc), CISC and RISC architectures.	13	13
4	Assembly Language Programming: Assembly language programs involving logical, branch and call instructions, sorting, evaluation of arithmetic expressions, string manipulation, assembler directives, procedures and macros.	15	13
5	Memory and Digital Interfacing: addressing and address decoding, interfacing RAM, ROM, EPROM, programmable peripheral interface, various modes of operation and interfacing to processor, interfacing keyboard, displays, etc.	11	11
	TOTAL	**64**	**60**

Specification Table Showing Distribution of Marks and Hours:

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	Structure of Computers	13	6	4	2	12
2	Micro Programmed Control	12	4	2	5	11
3	Introduction to Microprocessor Architecture	13	4	7	2	13
4	Assembly Language Programming	15	3	5	5	13
5	Memory and Digital Interfacing	11	4	4	3	11
	TOTAL	**64**	**22**	**22**	**16**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. Computer System Architecture, M. Moris Mano, Pearson/PHI, India.
2. Microprocessors Interface, Douglas V.Hall, Tata McGraw-Hill.
3. Computer Organization, Carl Hamacher, Zvonks Vranesic, SafeaZaky, McGraw-Hill
4. Advanced Microprocessors and Peripherals- Architecture, Programming and interfacing, A.K.Ray, K.M.Bhurchandi, Tata McGraw-Hill, New Delhi, India.
5. Computer Organization and Design: A Hardwar/Software Interface (MIPS Edition) by Patterson and Hennessy

Course Outcomes:

Have a good understanding of the functioning of a computer system as such and its various subcomponents. The student will be able to understand computing requirement for a specific purpose, analyse performance bottlenecks of the computing device and choose appropriate computing device for a given use case.

Algorithms								
Course code: COPC209				**Semester:** III				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Mid-Term Exam**	**Assignment**	**Attendance**	
3	1	0	4	60	20	15	5	100
Rationale / Course Learning Objectives: -								
The objective of this course is to prepare the student with the algorithmic foundations of computing. A sound grasp of algorithms is essential for any computer science engineer. Almost all programming involves algorithms at some level.								

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	**Fundamentals** Programming Models. Data Abstraction. Sets, Multisets, Stacks, Queues. Asymptotic and worst-case analysis of algorithms.	14	11
2	**Sorting** The sorting problem. Bubble sort, Selection sort, Insertion sort, Mergesort, Quicksort.	13	12
3	**Searching** Symbol Tables, Binary Search Trees, Balanced Search Trees. Hash Tables.	14	14
4	**Graphs** Definition of a directed and undirected graph. Paths, Cycles, spanning trees. Directed Acyclic Graphs. Topological Sorting. Minimum Spanning Tree algorithms. Shortest Path algorithms: Dijkstra's algo- rithm. Flow-based algorithms.	13	14
5	**Strings** String Sort. Tries. Substring Search. Regular Expressions. Elementary Data compression.	10	9
	TOTAL	**64**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	Fundamentals	14	5	3	3	11
2	Sorting	13	3	4	5	12
3	Searching	14	3	6	5	14
4	Graphs	13	6	3	5	14
5	Strings	10	4	3	2	9
	TOTAL	**64**	**21**	**19**	**20**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. Algorithms, Sedgewick and Wayne, Pearson
2. Introduction to Algorithms, Cormen, Leiserson, Rivest and Stein. MIT Press
3. Introduction to Theory of Computation, Sipser Michael, Cengage Learning.
4. Design & Analysis of Algorithms, Gajendra Sharma, Khanna Publishing House

Course Outcomes:

The student should be able to design basic algorithms for sorting and searching. The student should be able to understand the basic notions of time and space complexity of algorithms. The student should be able to implement sorting, searching, tree and graph algorithms in a modern computer programming language.

Computer Programming Lab							
Course code: COPC211				**Semester: III**			
Teaching Scheme				**Evaluation Scheme**			
Contact Hours/Week				**Practical**			
				End Term	**Progressive Assessment (PA)**		**Total Marks**
L	**T**	**P**	**Credit**		**Sessional**	**Viva Voice**	
0	0	4	2	40	40	20	100

Rationale / Course Learning Objectives: -

This Lab course is intended to practice what is taught in theory class of 'Computer Programming' and become proficient in computer programming. Computer programming is all about regular practice. Students should work on solved and unsolved problems listed in the text books, and the problems given by the teacher. Some of the topics that should necessary be covered in lab are listed below.

Course Content:

SI.NO	TOPICS FOR PRACTICE	HOURS
1	Familiarization with programming environment (Editor, Compiler, etc.)	3
2	Programs using I/O statements and various operators	4
3	Programs using expression evaluation and precedence	4
4	Programs using decision making statements and branching statements	4
5	Programs using loop statements	8
6	Programs to demonstrate applications of n dimensional arrays	6
7	Programs to demonstrate use of string manipulation functions	6
8	Programs to demonstrate parameter passing mechanism	6
9	Programs to demonstrate recursion	3
10	Programs to demonstrate use of pointers	6
11	Programs to demonstrate dynamic memory allocation	6
12	Programs to demonstrate file operations	4
	Total	60

Reference Books:

1. Let Us C, Yashavant Kanetkar
2. Problem Solving and Programming in C, R.S. Salaria, Khanna Publishing House
3. C Programming Absolute Beginner's Guide, Dean Miller and Greg Perry
4. The C Programming Language, Kernighan and Ritchie, Prentice Hall of India
5. Programming in ANSI C, E. Balagurusamy, Tata McGraw-Hill
6. C Programming & Data Structures, B. A. Fouruzan and R. F. Gilberg, CENGAGE Learning.

Course outcomes:

Student should be able to write code snippets, and then compile, debug and execute them.

Python Programming Lab									
Course code: COPC213				**Semester: III**					
Teaching Scheme				**Evaluation Scheme**					
Contact Hours/Week				**Practical**					
				End Term	**Progressive Assessment**			**Viva**	**Total Marks**
L	**T**	**P**	**Credit**		**Class Test**	**Assignment**	**Attendance**		
0	0	4	2	40	20	15	5	20	100

Rationale / Course Learning Objectives: -

The proposed course aims to provide students with hands-on experience in applying programming concepts to practical tasks. By focusing on coding exercises and projects, students will have the opportunity to reinforce their understanding of fundamental programming principles and develop problem-solving skills.

The course will cover a wide range of topics, including data types, control structures, functions, modules, file processing, regular expressions, SQL queries, and database interaction using DBI. Through these practical exercises, students will gain confidence in their ability to write efficient and effective code.

Course Content:

Sl. No.	UNIT	HOURS
1	Practice basic coding syntax	3
2	Write and execute scripts based on data types	6
3	Write and execute Python scripts with conditionals and loops	6
4	Write and execute Scripts based on Functions and Modules	8
5	File Processing scripts	8
6	Write and execute Regular Expressions	7
7	Write and execute SQL Queries	8
8	Write and execute scripts using DBI	8
9	Develop a simple web application	10
	TOTAL	**64**

Reference Books:

1. Taming Python by Programming, Jeeva Jose, Khanna Publishing House
2. Starting Out with Python, Tony Gaddis, Pearson
3. Core Python Programming, Wesley J. Chun, Prentice Hall

4. Python Programming: Using Problem Solving Approach, Reema Thareja, Oxford University Press
5. Introduction to Computation and Programming Using Python. John V. Guttag, MIT Press.
6. Beginning Python using Python 2.6 and Python 3, James Payne, Wrox publishing
7. Practical Programming: An Introduction to Computer Science using Python 3, Paul Gries, The Pragmatic Bookshelf

Course Outcomes:

At the end of the course student will be able to build program with a scripting language and will be able to learn any other scripting language on their own.

Data Structures Lab							
Course code: COPC215				**Semester: III**			
Teaching Scheme				**Evaluation Scheme**			
Contact Hours/Week				**Practical**			
				End Term	**Progressive Assessment (PA)**		**Total Marks**
L	**T**	**P**	**Credit**		**Sessional**	**Viva Voice**	
0	0	2	1	40	40	20	100

Rationale / Course Learning Objectives: -

This Lab course is intended to practice whatever is taught in theory class of 'Data Structures', 'Algorithms' and is an extension of previous course on 'Computer Programming'. Students should work on problems listed in the text books, and the problems given by the teacher. Some of the topics thatshould necessary be covered in lab are listed below. This Lab course requires a good coordination between theory course in Data Structures and Algorithms

Course Content:

SI.NO	TOPICS FOR PRACTICE	HOURS
1.	Write a program using recursive and non-recursive functions to perform search operation in a given list of integers using linear search technique	2
2.	Search operation in a given list of integers using binary search technique	2
3.	Write a program to implement insertion sorting for a given random data	2
4.	Write a program to implement bubble sorting for a given random data	2
5.	Write a program to implement quick sorting for a given random data	2
6.	Write a program to implement selection sorting for a given random data	1
7.	Write a program to implement heap sorting for a given random data	1
8.	Write a program to implement Hashing tables	2
9.	Write a program to implement single linked list	2
10.	Write a program to implement double linked list	2
11.	Write a program to implement circular linked list	2
12.	Write a program to Implement Stack operations using array and linked list	2
13.	Write a program to Implement Queue operations using array and linked list.	2
14.	Write a program to implement Breadth First Search (BFS)	2
15.	Write a program to implement Depth First Search (DFS)	2
16.	Write a program to implement a binary tree of integers	2
17.	Write a program to find the minimum depth of a binary tree	2
	Total	32

Reference Books:

1. Data Structures, R.S. Salaria, Khanna Book Publishing
2. Data Structures Using C, Reema Thareja, Oxford University Press India.
3. Classic Data Structures, Samanta Debasis, Prentice Hall of India.

4. Fundamentals of Data Structure in C, Horowitz, Ellis, Sahni, Sartaj, Anderson-Freed, Susan, University Press, India.
5. Data Structures: A Pseudo code approach with C, Richard F. Gilberg, Behrouz A. Forouzan, CENGAGE Learning, India.
6. Data Structures and Algorithms: Concepts, Techniques and Applications, G. A. V. Pai, McGraw- Hill Education, India.

Course Outcomes:

Student will be able to write programs for creating and doing different operations on various data structures. Student will be able to use/implement various algorithms learnt in the course on Algorithms. In summary student will have a good command over Data Structures and its applications in Algorithms.

Semester - IV

Operating Systems								
Course code: COPC202				**Semester:** IV				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Mid-Term Exam**	**Assignment**	**Attendance**	
3	0	0	3	60	20	15	5	100

Rationale / Course Learning Objectives: -

In this age of information technology computer system, is playing an important role for automation. Computers systems are used as an effective communication and decision-making tool for process and product automation in a business, industrial and educational environment. The technicians in the industry are expected to develop software systems.

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	**Introduction to Operating System** Overview of Operating System, basic concepts, UNIX/LINUX Architecture, Kernel, services and system calls, system programs	12	15
2	**Process Management** Process Concepts, operations on processes, IPC, Process Scheduling, Multithreaded programming	10	12
3	**File Management** Concept of a file, access methods, directory structure, file system mounting, file sharing and protection, file system structure and implementation, directory implementation, free-space management, efficiency and performance. Different types of file systems	10	12
4	**I/O System** Mass storage structure- overview, disk structure, disk attachment, disk scheduling algorithms, swap space management, RAID types.	8	11
5	**OS Security** Authentication, Access Control, Access Rights, System Logs	8	11
	TOTAL	**48**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	Introduction to Operating System	15	7	5	3	15
2	Process Management	10	5	4	3	12
3	File Management	10	5	4	3	12
4	I/O System	8	4	3	3	10
5	OS Security	8	5	3	3	11
	TOTAL	**48**	**26**	**19**	**15**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. Operating System Concepts, Silberschatz and Galvin, Wiley India Limited
2. UNIX Concepts and Applications, Sumitabha Das, McGraw-Hill Education
3. Operating Systems, Internals and Design Principles, Stallings, Pearson Education, India
4. Operating System Concepts, Ekta Walia, Khanna Publishing House
5. Modern Operating Systems, Andrew S. Tanenbaum, Prentice Hall of India
6. Operating systems, Deitel & Deitel, Pearson Education, India

Course Outcomes:

Students should be able to demonstrate basic knowledge about Operating System, be able to apply OS concepts such as processes, memory and file systems to system design, able to configure OS in an efficient and secure manner.

Introduction to DBMS								
Course code: COPC204				**Semester: IV**				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Mid-Term Exam**	**Assignment**	**Attendance**	
2	0	0	2	60	20	15	5	100

Rationale / Course Learning Objectives: -

The subject *Database Management System (DBMS)* is essential for efficiently storing, managing, and retrieving data. It covers data modeling, relational structures, SQL, normalization, and security, ensuring data integrity and optimization. Understanding DBMS enables students to design and implement robust databases, crucial for applications in software development, data analytics, and enterprise systems.

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	Introduction; Database System Concepts and Architecture	3	5
2	Data Modeling using the Entity-Relationship Model; The Enhanced Entity-Relationship (EER) model	5	10
3	The Relational Data Model and Relational Database Constraints; ER/EER to Relational Model mapping; Relational Algebra and Relational Calculus	8	15
4	SQL-99: Schema definition, Constraints, Queries, and Views; Security; Introduction to SQL programming Techniques	8	15
5	Functional dependencies and normalization for relational databases; Relational database design algorithms and further dependencies.	8	15
	TOTAL	**32**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	Introduction; Database System Concepts and Architecture	3	3	2	0	5
2	Data Modeling using the Entity-Relationship Model; The Enhanced Entity-Relationship (EER) model	5	2	3	5	10
3	The Relational Data Model and Relational Database Constraints; ER/EER to Relational Model mapping;Relational Algebra and Relational Calculus	8	2	3	10	15
4	SQL-99: Schema definition, Constraints, Queries, and Views; Security; Introduction to SQL programmingTechniques	8	5	5	5	15
5	Functional dependencies and normalization for relational databases; Relationaldatabase design algorithmsand further dependencies.	8	2	3	10	15
	TOTAL	**32**	**14**	**16**	**30**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. Fundamentals of Database Systems, Elmasri & Navathe, Pearson Education
2. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata McGraw-Hill.
3. Database System Concepts, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill, New Delhi, India.
4. Introduction to Database Systems, C.J.Date, Pearson Education
5. Introduction to SQL, Rick F.Vander Lans, Pearson Education

Course Outcomes:

After completing the course, the students will understand

1. How to design a database, database-based applications
2. How to use a DBMS
3. The critical role of databasesystem in designing several information system-based software systems or applications.

Computer Networks								
Course code: COPC206				**Semester:** IV				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		Mid Term Exam	**Assignment**	**Attendance**	
3	0	0	3	60	20	15	5	100

Rationale / Course Learning Objectives: -

The proposed course aims to provide students with a comprehensive understanding of computer networks, covering both theoretical concepts and practical applications. By exploring network topologies, models, transmission media, protocols, and devices, students will gain a solid foundation in network design, implementation, and management.

The course will delve into various layers of the OSI and TCP/IP models, allowing students to understand how different components of a network interact. Additionally, the course will cover practical aspects such as network devices, routing algorithms, and network management protocols.

Course Content:

Unit No.	UNIT	HOURS	MARKS
1	Introduction to computer networks; Network topologies; Network Models- OSI Reference Model, TCP/IP Model;	7	10
2	Transmission Media – principles, issues and examples; Wired Media – Coaxial, UTP, STP, Fiber Optic Cables; Wireless Media – HF, VHF, UHF, Microwave, Ku Band; Data Link Layer – design issues, example protocols (Ethernet, WLAN, Bluetooth); Switching Techniques;	21	26
3	Network Layer - design issues, example protocols (IPv4); Routing - principles/issues, algorithms, (Distance-vector, Link-state) and protocols (RIP, OSPF);	7	9
4	Transport Layer - design issues, example protocols (TCP); Application Layer Protocols (SMTP, DNS).	7	8
5	Functioning of Network Devices – NIC, Hub, Switch, Router, WiFi Devices; Network Management System and example protocol (SNMP).	6	7
	TOTAL	**48**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Marks
1	Introduction to computer networks	7	6	3	1	10
2	Transmission Media	21	16	6	4	26
3	Network Layer	7	6	2	1	9
4	Transport Layer	7	5	2	1	8
5	Functioning of Network Devices	6	4	2	1	**7**
	TOTAL	**48**	37	15	8	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. Computer Networks, 4th Edition (or later), Andrew S. Tanenbaum, PHI
2. TCP/IP Illustrated, Volume-1, W. Richard Stevens, Addision Wesley
3. Data and Computer Communications, William Stallings, PHI
4. An Engineering Approach to Computer Networking, S. Keshav, Addision Wesley/Pearson
5. An Integrated Approach to Computer Networks, Bhavneet Sidhu, Khanna Publishing House

Course Outcomes:

1. Understanding of computer networks, issues, limitations, options available.
2. Understanding of the care that needs to be taken while developing applications designed towork over computer networks
3. Able to configure basic LAN and connect computers to it.

Software Engineering								
Course code: COPC208				**Semester: IV**				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment (PA)**			**Total Marks**
L	**T**	**P**	**Credit**		**Mid-Term Exam**	**Assignment**	**Attendance**	
3	0	0	3	60	20	15	5	100
Rationale / Course Learning Objectives: -								
Inculcate essential technology and software engineering knowledge and skills essential to build a reasonably complex usable and maintainable software iteratively. 2) Emphasize on structured approach to handle software development. 3) Enhance communication skills.								

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	**Introduction to Software Engineering:** Lifecycle, Process Models - Traditional v/s Agile processes.	8	11
2	**Development Activities** Development Activities - Requirements Gathering and Analysis, Design Concepts, Software architecture and Architectural styles, Basic UI design, Effective Coding and Debugging techniques.	15	18
3	**Software Testing** Software Testing Basics, Unit, Integration, System and Acceptance Testing, Introduction to various testing techniques (e.g. Stress testing), Writing and executing test cases, Quality Assurance.	12	15
4	**Project Management** Project Management - Project management concepts, Configuration and Release Management, Version Control and its tools (Git), Release Planning, Change Management, Software Maintenance, Project Metrics.	13	16
	TOTAL	**48**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Marks
1	Introduction to Software Engineering	8	5	3	3	11
2	Development Activities	15	5	6	7	18
3	Software Testing	12	6	4	5	15
4	Project Management	13	7	6	3	16
	TOTAL	**48**				**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. SoftwareEngineering–APractitioner'sApproach,7thEdition,RogerPressman.
2. Softwareengineering,IanSommerville,PearsonEducation
3. AnIntegratedApproachtoSoftwareEngineering,PankajJalote,SpringerVerlag
4. SoftwareEngineering,NasibSinghGill,KhannaBookPublishingCo.India.
5. SoftwareEngineering,K.K.Agarval,YogeshSingh,NewAgeInternationalPublishers

Course Outcomes:

The proposed course is expected to provide an introduction to software engineering concepts and techniques to undergraduate students, thus enabling them to work in a small team to deliver a soft ware system. The course content and project will introduce various software technologies, process and project management skills that are needed for the delivery of software in a team setting.

Web Technologies								
Course code: COPC210				**Semester:** IV				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Mid-Term Exam**	**Assignment**	**Attendance**	
3	0	0	3	60	20	15	5	100
Rationale / Course Learning Objectives: -								
To provide basic skills on tools, languages and technologies related to website development. Learnings from this course may be used in the Mini Project and summer internship.								

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	**Fundamentals** Introduction to WWW, Web browser, web server, protocols-HTTP, POP, SMTP, FTP, WAP, secure connections, the web browser, domain name and registration process, web hosting, website and Web Application architecture (WAA).	4	6
2	**Introduction to HTML** Basic HTML tags, Basic Structure, Paragraph, Style Formatting, Hyperlinks, Images, List, Frames, Forms, and Tables.	8	10
3	**JavaScript** Client side scripting,;What is Javascript, simple Javascript, variables, functions, condition statements, loops and repetition	7	8
4	**Advance scripting** Javascript and objects; Javascript own objects, JS DOM and web browser environments, forms and validations. DHTML: What is DHTML, Advantages, disadvantages, difference between DHTML and HTML. AJAX: Introduction, advantages & disadvantages, AJAX based web application, Fetch API alternatives to AJAX XML, XSL and XSLT: Introduction to XML, uses of XML, simple XML, XML key components, DTD and Schemas, XML with application, XSL and XSLT.	16	20
5	**PHP** server side scripting: Introduction to PHP, syntax, variables, echo/print, Branching and looping statements ,Arrays, function and forms, advance PHP Databases :Basic command with PHP examples, Connection to server, creating database, selecting a database, listing	13	16

	database, listing table- names creating a table, inserting data, altering tables, queries, deleting database, deleting data and tables, PHP myadmin and database bugs.		
	TOTAL	**48**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	Introduction to WWW	4	4	2	0	6
2	Introduction to HTML	8	4	4	2	10
3	Javascript	7	2	4	4	8
4	Advance scripting	16	4	10	6	20
5	PHP	13	4	8	4	16
	TOTAL	**48**	**18**	**26**	**16**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. "Web Technologies--A Computer Science Perspective", Jeffrey C.Jackson,
2. "Internet & World Wide Web How To Program", Deitel, Deitel, Goldberg, Pearson Education
3. "Web programming- Building Internet Application", Chris Bales
4. Web Applications: Concepts and Real World Design, Knuckles.

Course Outcomes:

Student will be able to develop/build a functional website with full features

Operating Systems Lab							
Course code: COPC212				**Semester: IV**			
Teaching Scheme				**Evaluation Scheme**			
Contact Hours/Week				**Practical**			
				End Term	**Progressive Assessment (PA)**		**Total Marks**
L	**T**	**P**	**Credit**		**Sessional**	**Viva Voice**	
0	0	2	1	40	40	20	100

Rationale / Course Learning Objectives: -

This Lab course is intended to practice and do experiment on concepts taught in theory class of 'Operating Systems' and gain insight into functioning of the Operating Systems.

Course Content:

SI.NO	TOPICS FOR PRACTICE	HOURS
1	Revision practice of various commands like man, cp, mv, ln, rm, unlink, mkdir, rmdir, etcand many more that were learnt in IT Workshop course and later.	3
2	Implement two way process communication using pipes	3
3	Implement message queue form of IPC	2
4	Implement shared memory and semaphore form of IPC	2
5	Simulate the CPU scheduling algorithms - Round Robin, SJF, FCFS, priority	5
6	Simulate Bankers algorithm for Deadlock Avoidance and Prevention	3
7	Simulate all FIFO Page Replacement Algorithm using C program	3
8	Simulate all LRU Page Replacement Algorithms using C program	3
9	Simulate Paging Technique of Memory Management	3
10	Practice various commands/utilitiessuch as catnl, uniq, tee, pg, comm, cmp, diff, tr, tar, cpio, mount, umount, find, umask, ulimit, sort, grep, egrep,fgrep cut, paste, join, du, df , ps, who, etc and many more.	5
	Total	32

ReferenceBooks:

1. Operating System Concepts, Silberschatz, Abraham and Galvin, Peter, Wiley India Limited
2. UNIX Concepts and Applications, Sumitabha Das, McGraw-Hill Education
3. Operating System Concepts, Ekta Walia, Khanna Publishing House

Course Outcomes:

Students should be able to demonstrate basic knowledge about Operating System, be able to apply OS concepts such as processes, memory and file systems to system design, able to configure OS in anefficient and secure manner, and become an advance user.

Introduction to DBMS Lab							
Course code: COPC214				**Semester: IV**			
Teaching Scheme				**Evaluation Scheme**			
Contact Hours/Week				**Practical**			
				End Term	**Progressive Assessment (PA)**		**Total Marks**
L	**T**	**P**	**Credit**		**Sessional**	**Viva Voice**	
0	0	2	1	40	20	20	100

Rationale / Course Learning Objectives: -

The proposed course aims to provide students with practical experience in database design, implementation, and management. By focusing on real-world database applications, students will have the opportunity to apply their theoretical knowledge to solve complex problems.

The course will cover a variety of database scenarios, including employee databases, visitor management, student academics, inventory management, bank operations, and bus operator systems. Through these projects, students will gain experience in creating database schemas, designing tables, writing SQL queries, and implementing database constraints.

Course Content:

SI. No	TOPICS FOR PRACTICE	HOURS
1	Employee database – 'Create' employee table, 'Select' and display an employee matching a given condition, 'Delete' duplicate records, delete rows using triggers, insert and update records, find net salary, etc.	3
2	Visitor Management database	5
3	Students' Academic database	5
4	Inventory Management System database	6
5	Bank Operations database	6
6	Bus Operator (Roadways) – Do related activities such as prepare E-R Model, Relational Model, do Normalisation, Create Tables, Insert data, Delete Data, Query database, create stored procedures, etc.	7
	TOTAL	**32**

Reference Books:

1. Cisco press books on CCNA
2. User manual of networking devices available in the lab
3. Wiki pages on networking devices

Course Outcomes:

1. Understanding of computer networks, issues, limitations, options available.
2. Able to configure basic small LAN and connect computers to it.

Computer Networks Lab							
Course code: COPC216				**Semester: IV**			
Teaching Scheme				**Evaluation Scheme**			
Contact Hours/Week				**Practical**			
				End Term	**Progressive Assessment (PA)**		**Total Marks**
L	**T**	**P**	**Credit**		**Sessional**	**Viva Voice**	
0	0	2	1	40	40	20	100

Rationale / Course Learning Objectives: -

The proposed course aims to provide students with hands-on experience in networking fundamentals, focusing on cable management, device configuration, and network setup. By exploring different types of cables, connectors, devices, and configuration options, students will gain practical skills necessary for networking professionals.

The course will cover both hardware and software aspects of networking, allowing students to understand the physical components of a network as well as the configuration settings that govern network behavior. Through practical exercises, students will learn how to connect devices, troubleshoot network problems, and set up basic network topologies.

Course Content:

SI. No	TOPICS FOR PRACTICE	HOURS
1	Showing various types of networking cables and connectors, identifying them clearly.	1
2	Looking at specifications of cables and connectors of various companies on Internet, find out differences.	2
3	Making patch cords using different types of cables and connectors - crimping, splicing, etc	4
4	Demonstration of different type of cable testers, using them for testing patch cords pre-pared by the students in Lab and standard cables prepared by professionals	1
5	Configuring computing devices (PC, Laptop, Mobile, etc) for network, exploring different options and their impact – IP address, gateway, DNS, security options, etc	4
6	Showing various networking devices – NICs, Hub, Switch, Router, WiFi access point, etc.	1
7	Looking at specifications of various networking devices various companies on Internet,find out differences.	2
8	Network simulation tool (e.g. Cisco Packet Tracer)	5
9	Setting up a small wired LAN in the Lab	6
10	Setting up a small wireless LAN in the Lab	6
	Total	32

This is a skill course. More student practice will be better and try to find solution on their own.

Reference Books:

1. Cisco press books on CCNA
2. User manual of networking devices available in the lab
3. Wiki pages on networking devices

Course outcomes:

1. Understanding of computer networks, issues, limitations, options available.
2. Able to configure basic small LAN and connect computers to it.

Web Technologies Lab							
Course code: COPC218				**Semester:** IV			
Teaching Scheme				**Evaluation Scheme**			
Contact Hours/Week				**Practical**			
				End Term	**Progressive Assessment (PA)**		**Total Marks**
L	**T**	**P**	**Credit**		**Sessional**	**Viva Voice**	
0	0	4	2	40	40	20	100
Rationale / Course Learning Objectives: -							
This Lab course is intended to practice whatever is taught in theory class of 'Web Technologies'. Someof the things that should necessary be covered in lab are listed below							

Course Content:

SI.NO	TOPICS FOR PRACTICE	HOURS
1	Coding Server Client Programs	4
2	Developing Web Application using HTML, JavaScript	12
3	Developing Advanced Web Application Programs using CSS	4
4	Practicing PHP : Basics	4
5	Practicing PHP : Web Application Development	12
6	Practicing PHP: MySql - tiered Applications	12
7	Developing a fully functional Web Service Application using all the technologies learnedin this course.	12
	Total	60

Reference Books:

1. "Web Technologies--A Computer Science Perspective", Jeffrey C.Jackson,
2. "Internet & World Wide Web How To Program", Deitel, Deitel, Goldberg, Pearson Education
3. "Web programming- Building Internet Application", Chris Bales
4. Web Applications: Concepts and Real World Design, Knuckles

Course outcomes:

Student will be able to program web applications using and will be able to do the following:

1. Use LAMP Stack for web applications
2. Use Tomcat Server for Servlets and JSPs
3. Write simple applications with Technologies like HTML, Javascript, AJAX, PHP, Servlets and JSPs
4. Connect to Database and get results
5. Parse XML files using Java (DOM and SAX parsers)
6. Student will be able to develop/build a functional website with full features

Semester – V

Introduction to e-Governance								
Course code: COPC301				**Semester: V**				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Class Test**	**Assignment**	**Attendance**	
2	1	0	3	60	20	15	5	100
Rationale / Course Learning Objectives: -								
To cover the concept of e- Governance and to understand how technologies and business models shape the contours of government for improving citizen services and bringing in transparency.								

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	INTRODUCTION TO E-GOVERNANCE 1.1 Needs of e-Governance 1.2 Issues in e-Governance applications and the Digital Divide 1.3 Evolution of e-Governance. 1.4 Scope and content of e- Governance 1.5 Components of e-Governance 1.6 Present global trends of growth in e- Governance 1.7 Areas of e-Government 1.8 Critical success and failure factors for e-Governance 1.9 Role of social media in e-Governance	9	11
2	E-GOVERNANCE APPROACHES IN INDIA 2.1 Introduction to NeGP(National e-Governance Plan) 2.2 NeGP vision 2.3 The framework for e-Governance 2.4 National e-Governance strategy 2.5 Major components of national e-governance plan 2.6 Mission mode projects 2.7 Infrastructure pillars of NeGP. 2.8 Capacity building initiative under NeGP	9	11
3	E-GOVERNANCE PROJECT DEVELOPMENT AND MANAGEMENT 3.1 Introduction to e-Government project development 3.2 Conceptualization phase 3.3 Architecture phase,Define Phase,Support Phase 3.4 e-Government Project management Phase 3.5 Business Model for e-government project	10	13

	3.6 Public private partnership for e-Government 3.7 Security for e-Governance projects.		
4	CAPACITY BUILDING AND CHANGE MANAGEMENT 4.1 Capacity building for e-Governance 4.2 Governance structure for E-Governance projects 4.3 Change management for e-governance projects 4.4 Role of leadership in e-governance projects	10	12
5	GOVERNMENT PROCESS RE-ENGINEERING 5.1 Process reforms for e-Governance projects 5.2 Tools and techniques for Government process Re-engineering 5.3 Legal reforms 5.4 Technology management and Enterprise architecture for e-Eovernance 5.6 Case studies in e-Government (G2C,G2B)	10	13
	TOTAL	**48**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	**C**	**A**	**Total Mark**
1	Introduction to e-governance	9	3	4	4	11
2	E-governance approaches in india	9	3	4	4	11
3	E-governance project development and management	10	3	5	5	13
4	Capacity building and change management	10	3	5	4	12
5	Government process re-engineering	10	3	5	5	13
	TOTAL	48	15	23	22	60

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. C.S.R. Prabhu : E Governance: Concept and Case Studies, Prentice Hall Of India Pvt. Limited
2. Backus,Michiel : E Governance in Developing Countries ,IICD Research Brief No-1

Course Outcomes:

Through exposure to introductory ideas and practices followed in a selected number of e-Governance initiative in india,the course will help students to understand and appreciate the essence of e-Governance.

Internet of Things								
Course code: COPC303				**Semester: V**				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Class Test**	**Assignment**	**Attendance**	
2	1	0	3	60	20	15	5	100

Rationale / Course Learning Objectives: -

Internet of Things (IoT) is presently an important technology with wide ranging interest from Government, academia and industry. IoT cuts across different application domain verticals ranging from civilian to defense sectors which includes agriculture, space, health care, manufacturing, construction, water, mining, etc. Today it is possible to build different IoT solutions such as shopping system, infrastructure management in both urban and rural areas, remote health monitoring and emergency notification systems, and transportation systems. Therefore, it is very important to learn the fundamentals of this emerging technology.

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	Introduction to IoT, Defining IoT, Characteristics of IoT, Physical design of IoT, Logical design of IoT, Functional blocks of IoT.	10	12
2	IoT Communication Protocols: Explain the message passing protocols, MQTT, CoAP, XMPP, AMQP, DSS. Sensor networks, Logical Design of IoT, IoT functional blocks, IoT communication Models, IoT communication API's.	10	14
3	Sensors and Actuators: Sensors (working), PIR Motion Sensor, Sharp IR Distance Sensor, LDR Sensor, Gyro Sensor, Ultrasonic Distance Sensor, Need of ADC Chip while using Analog Sensors. Actuators (working): Servo Motor, Solenoid, Stepper Motor. Need of Relay while using Actuators.	10	13
4	Introduction to Arduino programming: Illustrate the Arduino Uno board, apply basic programming skills to develop the code for the Arduino Uno board, Practice the built- in library and user defined functions in the program, Interface different sensors and actuators with Arduino board.	12	15
5	Applications of IoT: Home automation, Industry applications, Surveillance applications, Health and Life Style, Wearable electronics and other IOT applications.	6	6
	TOTAL	**48**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	Introduction to IoT	10	4	4	4	12
2	IoT Communication Protocol	10	3	5	6	14
3	Sensors and Actuators	10	3	3	7	13
4	Introduction to Arduino programming	12	2	5	8	15
5	Application of IoT	6	2	0	4	6
	TOTAL	**48**	**14**	**17**	**29**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. "The Internet of Things: Enabling Technologies, Platforms, and Use Cases", by Pethuru Raj and Anupama C. Raman (CRC Press)
2. "The Internet of Things", by Samuel Greengard (The MIT Press)
3. "Internet of Things: Architecture and Design Principles", by Rajkamal (McGraw Hill)
4. "Internet of Things", by Dr. Jeeva Jose (Khanna Book Publishing)
5. "Designing the Internet of Things", by Adrian McEwen, Hakim Cassimally (John Wiley and Sons, Ltd.)
6. "The Fourth Industrial revolution", by Klaus Schwab (Portfolio Penguin)

Course Outcomes:

The student will develop underpinning knowledge, adequate design and programming skills competency for implementing various IoT applications using Arduino to attain the following course outcomes.

- Explain the basic concept of IoT
- Understand the basic concepts like usage of sensors, components and frequently used technologies of IoT from a global context
- Illustrate the application of Data protocols of IoT
- Understand various communication technologies of IOT
- Illustrate the use of sensor networks in applications of various domains

Mobile Computing								
Course code: COPE301				**Semester: V**				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Class Test**	**Assignment**	**Attendance**	
3	0	0	3	60	20	15	5	100

Rationale / Course Learning Objectives: -

The course teaches students about mobile computing, how to build Android mobile apps, covering UI design, app architecture and feature integration. Students are required to work on a project to develop a functional Android application, applying the skills they've learned.

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	**Mobile Computing** History of Mobile, Mobile generations, The Mobile Ecosystem, Types of Mobile Applications, Mobile Information Architecture Android Versions, Features of Android, Android Architecture, Installing Android SDK Tools, Configuring Android in Eclipse IDE, Android Development Tools (ADT), Android Virtual Devices (AVD).	**6**	**7**
2	**Android Fundamentals** Creating first android application, Anatomy of android application, Deploying Android app (USB interface), Android application components, Android Activity, Activity life cycle, Intent objects, Intent Types, Intents Activity Linking.	**8**	**14**
3	**Android UI/UX Components** Android Fragments - Life cycle, Interaction between fragments, List fragment, Dialog fragment. Understanding the components of a screen (Layouts), Adaptive Display Orientation, Action Bar, Views (UI Widgets) - Button, Toast, ToggleButton, CheckBox, RadioButton, Spinner, WebView, EditText, DatePicker, TimePicker, ListView, ProgressBar. Analog and Digital clock, UI Event Handling.	**9**	**16**

4	**Advanced Android Components** Menus-Option, Android Context, Popup, Images-ImageView, ImageSwitcher, AlertDialog, Alarm manager, SMS, E-mail, Media Player, Camera API – Image, video. Telephony Manager.	**11**	**10**
5	**Android Data Handling and Publishing** Persistent Data- Data Storage Options: preferences, Internal Storage, External Storage, Content Provider. SQLite database - Connecting database and operations-Insert, Delete, Update, Fetch. Publishing android applications, Deploying APK files.	**14**	**13**
	TOTAL	**48**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	Mobile Computing	6	3	2	2	7
2	Android Fundamentals	8	4	4	6	14
3	Android UI/UX Components	9	5	4	7	16
4	Advanced Android Components	11	2	3	5	10
5	Android Data Handling and Publishing	14	3	4	6	13
	TOTAL	**48**	**17**	**17**	**26**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. Pradeep Kothari, "Android Application Development Black Book", DreamTech Press.
2. Wei-Meng Lee, Beginning Android 4 Application Development, Wiley Publishing, Inc.
3. Sayed Y Hashimi and Satya Komatineni(2009), "Pro Android", Wiley India Pvt Ltd.
4. Mark L Murphy, "Beginning Android", Wiley India Pvt Ltd.
5. Reto Meier, Professional Android 4 Application Development, Wiley India Pvt Ltd.
6. James C.Sheusi, "Android Application Development for Java Programmers", Cengage Learning.

Course Outcomes:

Upon completion of the course, students will be able to:

1. Develop basic mobile applications for the Android platform.
2. Utilize Android development tools, such as Android Studio, to develop android apps.
3. Integrate various Android components and services, including activities, fragments, and background tasks, into their applications.

Information Security								
Course code: COPE303				**Semester: V**				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Class Test**	**Assignment**	**Attendance**	
3	0	0	3	60	20	15	5	100
Rationale / Course Learning Objectives: -								
To learn how to evaluate and enhance information security of IT infrastructure and organizations.								

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	**Introduction to Information Security** 1.1 Various aspects of information security (PAIN) 1.2 Types of attack 1.3 Security Features of Operating Systems – Authentication, Logs, Audit Features, File System Protection, User Privileges, RAID options, Anti-Virus Software.	7	9
2	**Understanding security weaknesses** 2.1 Security Weaknesses in popular networking protocols – IP, TCP, UDP, RIP, OSPF, HTTP, SMTP 2.2 Security weaknesses in common networking devices – Hub, switch, router, WiFi 2.3 Security solutions to mitigate security risk of networking protocols (IPSec, HTTPS) and devices (VLAN, VPN, Ingress Filtering)	8	8
3	**Cryptography** 3.1 Substitution Techniques, Transposition Techniques 3.2 Symmetric and Asymmetric Key Cryptography- Overview of Symmetric Key Encryption, Data Encryption Standard, Overview of Asymmetric Key Encryption, RSA Algorithm, Digital Signature, Digital Certificate. 3.3 Public Key Infrastructure (PKI) 3.4 Security considerations while developing software	15	20
4	**Network Security Products** 4.1 Firewall- ACL, Packet Filtering, DMZ, Alerts and Audit Trials 4.2 IDS/IPS and its types (Signature based, Anomaly Based, Policy Based) 4.3 VPN Concentrator and Content Screening Gateways.	10	15
5	**Introduction to Security Standards** 5.1 ISO 27001, Indian IT Act, IPR Laws 5.2 Security Audit procedures 5.3 Developing Security Policies	8	8

	5.4 Disaster Recovery, Business Continuity Planning		
	TOTAL	**48**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	**C**	**A**	**Total Mark**
1	Introduction to Information Security	7	4	2	3	9
2	Understanding security weaknesses	8	3	2	3	8
3	Cryptography	15	3	7	10	20
4	Network Security Products	10	4	5	6	15
5	Introduction to Security Standards	8	4	2	2	8
	TOTAL	**48**	**18**	**18**	**24**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. Information Security and Cyber Laws, Sarika Gupta, Khanna Publishing House
2. RFCs of protocols listed in content (https://www.ietf.org)
3. Various Acts, Laws and Standards (IT Act, ISO27001 Standard, IPR and Copyright Laws, etc.)
4. Security Guideline documents of Operating Systems (OS Manual, Man Pages, etc)
5. https://www.cert-in.org.in/
6. https://www.sans.org/

Course Outcomes:

Understanding of security needs and issues of IT infrastructure. Have basic skills on security audit of networks, operating systems and application software.

Mobile Computing Lab							
Course code: COPE305				**Semester: V**			
Teaching Scheme				**Evaluation Scheme**			
Contact Hours/Week				**Practical**			
				End Term	**Progressive Assessment (PA)**		**Total Marks**
L	**T**	**P**	**Credit**		**Sessional**	**Viva Voce**	
0	0	2	1	40	40	20	100
Rationale / Course Learning Objectives: -							
The course teaches students about mobile computing, how to build Android mobile apps, covering UI design, app architecture and feature integration. Students are required to work on a project to develop a functional Android application, applying the skills they've learned.							

Course Content:

SI.NO	TOPICS FOR PRACTICE	HOURS
1	Setting up the Android IDE and AVD.	2
2	Create a Basic Android Hello World App.	4
3	Create an app with multiple Activities and link them with intents.	2
4	Create a Survey form using Android UI Widgets.	4
5	Develop a simple timer app with Start, Pause, and Reset buttons.	4
6	Create a simple Phone dialler to make calls.	4
7	Create an Image search App using a 3rd Party API.	4
8	Create a Simple Notepad with Create, Read, Update and Delete operations with Persistent Data.	6
9	Generate signed APK files for all the above Experiments and Share them.	2
	Total	32

Reference Books:

1. Pradeep Kothari, “Android Application Development Black Book”, DreamTech Press.
2. Wei-Meng Lee, Beginning Android 4 Application Development, Wiley Publishing, Inc.
3. Sayed Y Hashimi and Satya Komatineni(2009), “Pro Android”, Wiley India Pvt Ltd.
4. Mark L Murphy, “Beginning Android”, Wiley India Pvt Ltd.
5. Reto Meier, Professional Android 4 Application Development, Wiley India Pvt Ltd.
6. James C.Sheusi, “Android Application Development for Java Programmers”, Cengage Learning.

Course Outcomes:

Upon completion of the course, students will be able to:

1. Develop basic mobile applications for the Android platform.
2. Utilize Android development tools, such as Android Studio, to develop android apps.
3. Integrate various Android components and services, including activities, fragments, and background tasks, into their applications.

Information Security Lab							
Course code: COPE307				**Semester: V**			
Teaching Scheme				**Evaluation Scheme**			
Contact Hours/Week				**Practical**			
				End Term	**Progressive Assessment (PA)**		**Total Marks**
L	**T**	**P**	**Credit**		**Sessional**	**Viva Voce**	
0	0	2	1	40	40	20	100
Rationale / Course Learning Objectives: -							
This Lab course is intended to practice whatever is taught in theory class of 'Information Security' and learn how to evaluate and enhance information security of IT infrastructure and organizations.							

Course Content:

SI.NO	TOPICS FOR PRACTICE	HOURS
1	a) Install and configure Antivirus software on system b) Set up operating system Updates	4
2	Perform Backup and Restore of the system	3
3	Set up passwords to operating system and applications.	3
4	Apply security to file folder or application using access permissions and verify	3
5	Write a program to implement Caesar Cipher	3
6	Write a program to implement Vernam Cipher	3
7	Write a program to implement Rail Fence Technique	3
8	Write a program to implement simple columnar transposition technique	3
9	a) Install firewall on any operating system b) Configure firewall settings on any operating system	4
10	Trace the path of web site using Tracert Utility	3
	Total	32

Course Outcomes:

Understanding of security needs and issues of IT infrastructure. Have basic skills on security audit of networks, operating systems and application software.

Semester – VI

Multimedia Technologies								
Course code: COPE302				**Semester: VI**				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Class Test**	**Assignment**	**Attendance**	
3	0	0	3	60	20	15	5	100
Rationale / Course Learning Objectives: -								
To introduce students to the domain of Multimedia Technologies, which explain the technologies underlying digital images, videos and audio contents, including various compression techniques and standards, and the issues to deliver multimedia content over the Internet.								

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	**Introduction to Multimedia** 1.1 Components of Multimedia 1.2 Application of Multimedia 1.3 Multimedia Hardware and Software 1.4 Multimedia Operating Systems 1.5 Multimedia Communication System	5	7
2	**Basic Compression Techniques** 2.1 Lossless compression algorithms: Introduction, Basics of Information Theory, Run Length Encoding, Variable-Length Coding (Huffman Coding, Adaptive Huffman Coding), Dictionary Based Coding (LZW). 2.2 Lossy compression algorithms: Introduction, Distortion Measures, The Rate Distortion Theory, Quantization, Transform Coding (Discrete Cosine Transform), Wavelet-Based Coding (Continuous Wavelet Transform, Discrete Wavelet Transform, JPEG. 2.3 Audio and Video Compression Techniques: Introduction to audio and Video Compression, H.261, H.265, MPEG, MP3, MP4, FLAC, ALAC, ITU G.722.	14	15
3	**Content Development and Distribution** 3.1 Desktop Publishing: Introduction to Desktop Publishing, Merits & Demerits of Desktop Publishing, Design Principles of Desktop Publishing. 3.2 Photoshop: Introduction to Photoshop, Application of Photoshop, Different versions, workspace, tools, Photo editing/inserting, Palettes, layers, Digital Image, pixels,	12	15

	resolution, DPI, raster images/bitmaps, vector images/graphics, various file formats: PSD, JPEG, GIF, TIF, PNG. 3.3 Corel Draw: Introduction to Corel Draw, Features of Corel Draw, Corel Draw Interface, Different Tools in Corel Draw, types of text. 3.4 Page Maker: Introduction to Page Maker, Different versions, Concepts and Applications of PageMaker guides and rulers, Drawing tools, fills and outline. 3.5 Multimedia Animation & Special Effects: Introduction Animation, Types of Animation, Uses of Animation, 2D & 3D animation, Principle of Animation, File Formats. 3.6 Flash: Introduction to Flash, Different tools in Flash, Types of animation/ tweening in Flash, Timeline		
4	**Introduction to Digital Imaging** 4.1 Graphics Design: Introduction to Graphic Design, Implications and Impact of Graphic Design 4.2 Principles and Elements of Graphics Design: Introduction to Drawing, Types of Drawing, Color theories, Color Wheel, Fundamentals Visuals Compositions, Typography, Principles of Layout Design and use of Digital technology 4.3 Digital Image: Definition of Digital images, Digital imaging in multimedia- Types of Images, Representation of digital image, Phases of image processing, Elements of digital image processing systems, Color Models.	8	11
5	**Introduction to Multimedia Programming and Applications** 5.1 Abstractions Levels of Multimedia Programming-Libraries, System Software, Toolkits, Higher Programming Languages, Object Oriented approaches. 5.2 Multimedia Applications- Media preparation and composition, Media integration and communication, Media Entertainment.	9	12
	TOTAL	**48**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	Introduction to Multimedia	5	2	3	2	7
2	Basic Compression Techniques	14	5	4	6	15
3	Content Development and Distribution	12	4	5	6	15
4	Introduction to Digital Imaging	8	4	3	4	11
5	Introduction to Multimedia Programming and Applications	9	3	3	6	12
	TOTAL	**48**	**18**	**18**	**24**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

1. An Introduction to Multimedia Authoring, A. Eliens
2. Multimedia: Computing, Communications and Applications, Ralf Steinmetz and Klara Nahrstedt, Pearson Education Asia
3. Fundamentals of Multimedia, Prentice Hall/Pearson, Ze-Nian Li & Mark S. Drew.
4. Multimedia and Animation, V.K. Jain, Khanna Publishing House, Edition 2018
5. Fundamentals of Multimedia, Ramesh Bangia, Khanna Book Publishing Co., N. Delhi (2007)

Course Outcomes:

Student will understand various aspects of Multimedia and related standards. Student will be able to build multimedia content and applications and also multimedia enables Web applications and mobile applications

FOSS								
Course code: COPE304				**Semester: VI**				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Class Test**	**Assignment**	**Attendance**	
3	0	0	3	60	20	15	5	100

Rationale / Course Learning Objectives: -

This course will provide the learner with comprehensive exposure to the philosophy, principles, and tools associated with Free and Open-Source Software (FOSS). Throughout this course, students will gain a deep understanding of the fundamental concepts and benefits of FOSS, as well as practical skills in utilizing various FOSS tools effectively.

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	**FOSS PHILOSOPHY** Understanding the FOSS Community and FOSS Philosophy, Benefits of Community based Software Development, Guidelines for working with FOSS community, Requirements for being open, free software, open-source software, FOSS Licensing Models, FOSS example.	**8**	**8**
2	**LINUX** Linux Installation and Hardware Configuration, Boot Process, Dual-Booting Linux and other Operating Systems, Kernel Options during Boot, X Windows System Configuration, System Administration (Server Administration, Backup and Restore Procedures, Strategies for keeping a Secure Server).	**12**	**15**
3	**Programming Tools and Techniques** LibreOffice Tools; Samba: Cross platform; Introduction about LAMP. Brief Introduction to Programming using languages like Java /Python / Perl. Database Systems MySQL, PostgreSQL or equivalent. Open-Source UML Tools; Introduction to Mobile Programming; Version Control Systems like SVN, Git or equivalent; Project Management Tools; Bug Tracking Systems; Package Management Systems.	**18**	**27**
4	**FOSS CASE STUDIES** Some example case studies of FOSS implementation.	**10**	**10**
	TOTAL	**48**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	FOSS philosophy	8	5	2	1	8
2	Linux	12	4	5	6	15
3	Programming Tools and Techniques	18	7	7	13	27
4	FOSS case studies	10	2	4	4	10
	TOTAL	**48**	**18**	**18**	**24**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

References:

1. Linux in a Nutshell, by Ellen Siever
2. Philosophy of GNU URL: http://www.gnu.org/philosophy/.
3. Linux Administration URL: http://www.tldp.org/LDP/lame/LAME/linux-admin-madeeasy/.
4. Version control system URL: http://git-scm.com/.
5. Samba: URL: http://www.samba.org/.
6. Libre office: http://www.libreoffice.org/.

Course Outcomes:

1. Student will be able to work with FOSS tools.
2. Find and evaluate FOSS alternatives for any software requirement.

Multimedia Technologies Lab							
Course code: COPE306				**Semester: VI**			
Teaching Scheme				**Evaluation Scheme**			
Contact Hours/Week				**Practical**			
				End Term	**Progressive Assessment (PA)**		**Total Marks**
L	**T**	**P**	**Credit**		**Sessional**	**Viva Voce**	
0	0	2	1	40	40	20	100
Rationale / Course Learning Objectives: -							
This Lab course is intended to practice whatever is taught in theory class of 'Multimedia Technologies'. Students should explore features of various tools introduced during the course and become comfortable.							

Course Content:

SI.NO	TOPICS FOR PRACTICE	HOURS
1	To study multimedia hardware system and basic software tools	2
2	Photoshop: 2.1Scanning image 2.2 Creating new images 2.3 Changing foreground and background colors 2.4 Creating and using paths 2.5 Editing and retouching 3.6 Duplicating images 2.6 Layers - linking with layers 2.7 Grouping a images 2.8 Rubber stamp and pattern stamp tool 2.9 Painting - paintbrush tool, air-brush tool, pencil tool, eraser tool, gradient tools 2.10 Photoshop filters	8
3	Corel Draw: 3.1 Drawing lines, shapes, objects, tables, templates. 3.2 Importing and exporting files 3.3 Make Business card 3.4 Logo Design	7
4	Page Maker: 4.1 Creating and opening a document in Page maker 4.2 Formatting and editing a document 4.3 Drawing and shaping objects 4.4 Insertion of text and graphics in a given document from external source 4.5 Using columns utility to give the document column look.	7
5.	Flash: 5.1 Flash document setup 5.2 Create animation	8

	5.3 Transforming text, Special effects: Shape tween, Mask, Spotlight, Motion Guides, Motion Tweens.	
	Total	32

Course Outcomes:
Student will understand various aspects of Multimedia and related standards. Student will be able to build multimedia content and applications

FOSS Lab							
Course code: COPE308				**Semester: VI**			
Teaching Scheme				**Evaluation Scheme**			
Contact Hours/Week				**Practical**			
				End Term	**Progressive Assessment (PA)**		**Total Marks**
L	**T**	**P**	**Credit**		**Sessional**	**Viva Voce**	
0	0	2	1	40	40	20	100

Rationale / Course Learning Objectives: -

This course will provide the learner with comprehensive exposure to the philosophy, principles, and tools associated with Free and Open-Source Software (FOSS). Throughout this course, students will gain a deep understanding of the fundamental concepts and benefits of FOSS, as well as practical skills in utilizing various FOSS tools effectively.

Course Content:

SI.NO	TOPICS FOR PRACTICE	HOURS
1	Installing a Linux distribution (e.g., Ubuntu or Fedora) on a virtual machine.	2
2	Creating and formatting documents using LibreOffice.	2
3	Create Simple Application Using Lamp Stack.	6
4	Writing Simple Programs in Java, Python and Perl.	4
5	Creating MySQL databases and tables, and performing basic CRUD (Create, Read, Update, Delete) operations.	4
6	Installing packages using package managers.	2
7	Setting up and configuring a simple web server using Apache or Nginx	4
8	Develop a Basic Android App in Android Studio.	4
9	Setup a Git Repository for a mini project and perform basic operations.	4
	Total	32

References:

7. Linux in a Nutshell, by Ellen Siever
8. Philosophy of GNU URL: http://www.gnu.org/philosophy/.
9. Linux Administration URL: http://www.tldp.org/LDP/lame/LAME/linux-admin-madeeasy/.
10. Version control system URL: http://git-scm.com/.
11. Samba: URL: http://www.samba.org/.
12. Libre office: http://www.libreoffice.org/.

Course Outcomes:

3. Student will be able to work with FOSS tools.
4. Find and evaluate FOSS alternatives for any software requirement.

Artificial Intelligence								
Course code: COOE302				**Semester: VI**				
Teaching Scheme				**Evaluation Scheme**				
Contact Hours/Week				**Theory**				
				End Term	**Progressive Assessment**			**Total Marks**
L	**T**	**P**	**Credit**		**Class Test**	**Assignment**	**Attendance**	
3	0	0	3	60	20	15	5	100

Rationale / Course Learning Objectives: -

Artificial Intelligence (AI) is a rapidly growing field with profound implications for various industries and aspects of human life. Understanding AI is essential for individuals and organizations to stay competitive and adapt to the technological advancements of the 21st century. This course aims to provide students with a strong foundation in AI principles, techniques, and applications. By studying AI, students will develop the skills necessary to analyze complex problems, design intelligent systems, and contribute to the development of innovative solutions.

Course Content:

UNIT NO	UNIT	HOURS	MARKS
1	**Introduction to Artificial Intelligence** 1. Artificial Intelligence (AI) definition 2. Goals of AI 3. History of AI 4. Applications of AI	7	10
2	**Agents and Environments** 1. Agent Terminology, Types of Agents – Simple Reflex Agents, Model Based Reflex Agents, Goal Based Agents 2. Nature of Environments, Properties of Environments	9	10
3	**Search Algorithms Terminology** 1. Brute Force Search Strategies – Breadth First Search, Depth First Search. 2. Heuristic Search Strategies, Local Search Algorithms.	9	12
4	**Fuzzy Logic Systems** 1. Introduction to Fuzzy Logic and Fuzzy systems, 2. Membership functions, 3. Fuzzification/Defuzzification	12	14
5	**Neural Networks** 1. Basic structure of Neural Networks 2. Perceptron 3. Back-propagation	11	14
	TOTAL	**48**	**60**

Specification Table Showing Distribution of Marks and Hours

Unit No	UNIT	HOURS	Marks			
			K	C	A	Total Mark
1	Introduction to Artificial Intelligence	7	4	3	3	10
2	Agents and Environments	9	3	4	3	10
3	Search Algorithms Terminology	9	2	4	6	12
4	Fuzzy Logic Systems	12	3	4	7	14
5	Neural Networks	11	3	4	7	14
	TOTAL	**48**	**15**	**19**	**26**	**60**

Abbreviations: **K**=Knowledge level, **C**= Comprehension Level, **A**=Application level

Reference Books:

5. Russell, Norvig, Artificial intelligence: A modern approach, 2nd edition. Pearson/Prentice Hall.
6. M.C. Trivedi, A classical approach to Artificial Intelligence, Khanna Publishing House, New Delhi (2018)
7. Konstantin V Titov, Introduction to Artificial Intelligence: Understanding the Basics (2024)
8. Ethem Alpaydin, Introduction to Machine Learning, Second Edition, http://mitpress.mit.edu/catalog/item/default.asp?ttype=2&tid=12012.

Course Outcomes:

At the end of the course, the student will be able to:

CO1	Define artificial intelligence and its goals.
CO2	Differentiate between different types of agents and understand the characteristics and properties of various environments.
CO3	Implement brute force search strategies (BFS, DFS) and heuristic search strategies.
CO4	Explain the principles of fuzzy logic and fuzzy systems and understand the concept of membership functions and the processes of fuzzification and defuzzification.
CO5	Describe the basic structure of neural networks and understand the concept of a perceptron and the backpropagation algorithm.