



PERIYAR UNIVERSITY

PERIYAR PALKALAINAGAR SALEM-

636011

DEGREE OF BACHELOR OF SCIENCE

Syllabus for

B.Sc., COMPUTER SCIENCE

(SEMESTER PATTERN - CBCS)

(For Candidates admitted in the colleges affiliated to Periyar

University from 2026-2027 onwards)

1.Introduction

B.Sc.ComputerScience

Education is the key to development of any society. Role of higher education is crucial for securing right kind of employment and also to pursue further studies in best available world class institutes elsewhere within and outside India. Quality education in general and higher education in particular deserves high priority to enable the young and future generation of students to acquire skill, training and knowledge in order to enhance their thinking, creativity, comprehension and application abilities and prepare them to compete, succeed and excel globally. Learning Outcomes-based Curriculum Framework (LOCF), which makes it student-centric, interactive and outcome-oriented with well-defined aims, objectives and goals to achieve. LOCF also aims at ensuring uniform education standard and content delivery across the state which will help the students to ensure similar quality of education irrespective of the institute and location.

Computer Science is the study of quantity, structure, space and change, focusing on problem solving, application development with wider scope of application in science, engineering, technology, social sciences etc. throughout the world in last couple of decades and it has carved out a space for itself like any other disciplines of basic science and engineering. Computer science is a discipline that spans theory and practice and it requires thinking both in abstract terms and in concrete terms. Nowadays, practically everyone is a computer user, and many people are even computer programmers. Computer Science can be seen on a higher level, as a science of problem solving and problem solving requires precision, creativity, and careful reasoning. The ever-evolving discipline of computer science also has strong connections to other disciplines. Many problems in science, engineering, health care, business, and other areas can be solved effectively with computers, but finding a solution requires both computer science expertise and knowledge of the particular application domain. Computer science has a wide range of specialties. These include Computer Architecture, Software Systems, Graphics, Artificial Intelligence, Computational Science, and Software Engineering. Drawing from a common core of computer science knowledge, each specialty area focuses on specific challenges. Computer Science is practiced by mathematicians, scientists and engineers. Mathematics, the origin of Computer Science, provides reason and logic. Science provides the

methodology for learning and refinement. Engineering provides the techniques for building hardware and software.

1. Programme Outcome, Programme Specific Outcome and Course Outcome

Computer Science is the study of quantity, structure, space and change, focusing on problem solving, application development with wider scope of application in science, engineering, technology, social sciences etc. The key core areas of study in Mathematics include Algebra, Analysis (Real & Complex), Differential Equations, Geometry, and Mechanics. The Students completing this programme will be able to present Software application clearly and precisely, make abstract ideas precise by formulating them in the Computer languages. Completion of this programme will also enable the learners to join teaching profession, enhance their employability for government jobs, jobs in software industry, banking, insurance and investment sectors, data analyst jobs and jobs in various other public and private enterprises.

LEARNING OUTCOMES-BASED CURRICULUM FRAMEWORK GUIDELINES BASED REGULATIONS FOR UNDER GRADUATE PROGRAMME	
Programme:	B.Sc., Computer Science
Programme Code:	
Duration:	3 years [UG]
Programme Outcomes:	PO1: Disciplinary knowledge: Capable of demonstrating comprehensive knowledge and understanding of one or more disciplines that form a part of an undergraduate Programme of study. PO2: Communication Skills: Ability to express thoughts and ideas effectively in writing and orally; Communicate with others using appropriate media; confidently share one's views and express herself/himself; demonstrate the ability to listen carefully, read and write analytically, and present complex information in a clear and concise manner to different groups. PO3: Critical thinking: Capability to apply analytic thought to a body of knowledge; analyze and evaluate evidence, arguments, claims, beliefs on the basis of empirical evidence; identify relevant assumptions or implications; formulate coherent arguments; critically evaluate practices, policies and theories by following scientific approach to knowledge development. PO4: Problem solving: Capacity to extrapolate from what one has learned and apply their competencies to solve different kinds of non-familiar problems, rather than replicate curriculum content knowledge; and apply one's learning to real life situations. PO5: Analytical reasoning: Ability to evaluate the reliability and relevance of evidence; identify logical flaws and holes in the arguments of others; analyze and synthesize data from a variety of sources; draw valid conclusions and support them with the evidence and examples, and address opposing viewpoints. PO6: Research-related skills: A sense of inquiry and capability for asking relevant/appropriate questions, problem arising, synthesizing and articulating; Ability to recognize cause-and-effect relationships, define problems, formulate hypotheses, test hypotheses, analyses, interpret and draw conclusions from data, establish hypotheses, predict cause-and-effect relationships; ability to plan,

	execute and report the results of an experiment or investigation. PO7: Cooperation/Team work: Ability to work effectively and respectfully with diverse teams; facilitate cooperative or coordinated effort on the part of a group, and act together as a group or a team in the interests of a common cause and work efficiently as a member of a team PO8: Scientific reasoning: Ability to analyse, interpret and draw conclusions from quantitative/qualitative data; and critically evaluate ideas, evidence and experiences from an open-minded and reasoned perspective. PO9: Reflective thinking: Critical sensibility to lived experiences, with self-awareness and reflexivity of both self and society. PO10 Information/digital literacy: Capability to use ICT in a variety of learning situations, demonstrate ability to access, evaluate, and use a variety of relevant information sources; and use appropriate software for analysis of data. PO11 Self-directed learning: Ability to work independently, identify appropriate resources required for a project, and manage a project through to completion. PO12 Multicultural competence: Possess knowledge of the values and beliefs of multiple cultures and a global perspective; and capability to effectively engage in a multicultural society and interact respectfully with diverse groups. PO13: Moral and ethical awareness/reasoning: Ability to embrace moral/ethical values in conducting one's life, formulate a position/argument about an ethical issue from multiple perspectives, and use ethical practices in all work. Capable of demonstrating the ability to identify ethical issues related to one's work, avoid unethical behaviors such as fabrication, falsification or misrepresentation of data or committing plagiarism, not adhering to intellectual property rights; appreciating environmental and sustainability issues; and adopting objective, unbiased and truthful actions in all aspects of work. PO14: Leadership readiness/qualities: Capability for mapping out the tasks of a team or an organization, and setting direction, formulating an inspiring vision, building a team who can help achieve the vision, motivating and inspiring team members to engage with that vision, and using management skills to guide people to the right destination, in a smooth and efficient way. PO 15: Lifelong learning: Ability to acquire knowledge and skills, including "learning how to learn", that are necessary for participating in learning activities throughout life, through self-paced and self-directed learning aimed at personal development, meeting economic, social and cultural objectives, and adapting to changing trades and demands of work place through knowledge/skill development/reskilling.
--	---

2. Highlights of the Revamped Curriculum

- Student-centric, meeting the demands of industry & society, incorporating industrial components, hands-on training, skill enhancement modules, industrial project, project with viva-voce, exposure to entrepreneurial skills, training for competitive examinations, sustaining the quality of the core components and incorporating application oriented content wherever required.

- The Core subjects include latest developments in the education and scientific front, advanced programming packages allied with the discipline topics, practical training, devising mathematical models and algorithms for providing solutions to industry/real life situations. The curriculum also facilitates peer learning with advanced mathematical topics in the final semester, catering to the needs of stakeholders with research aptitude.
- The General Studies and Computer Science based problem solving skills are included as mandatory components in the 'Training for Competitive Examinations' course at the final semester, a first of its kind.
- The curriculum is designed so as to strengthen the Industry-Academia interface and provide more job opportunities for the students.
- The Industrial Statistics course is newly introduced in the fourth semester, to expose the students to real life problems and train the students on designing a mathematical model to provide solutions to the industrial problems.
- The Internship during the second year vacation will help the students gain valuable work experience that connects classroom knowledge to real world experience and to narrow down and focus on the career path.
- Project with viva-voce component in the fifth semester enables the student, application of conceptual knowledge to practical situations. The state of art technologies in conducting a Explain in a scientific and systematic way and arriving at a precise solution is ensured. Such innovative provisions of the industrial training, project and internships will give students an edge over the counterparts in the job market.
- State-of Art techniques from the streams of multi-disciplinary, cross disciplinary and inter disciplinary nature are incorporated as Elective courses, covering conventional topics to the latest – Statistics with R Programming, Data Science, Machine learning, Internet of Things and Artificial Intelligence etc..

DURATION OF THE PROGRAMME

The programme shall extend over a period of three years comprising six semesters with two semesters in one academic year. There shall not be less than 90 working days for each semester. Examination shall be conducted at the end of every semester for the respective subjects.

COURSE OF STUDY

The programme of study shall comprise instruction in the following subjects according to the syllabus and books prescribed from time to time. The syllabus for various subjects shall be clearly demarcated into five units in each subject. Part -I, Part-II, Part- III and Part -IV subjects are prescribed in the scheme of examination. The Extension Activities are a must for each student to take part at least in any one of the activities such as NSS, YRC, SPORTS and RRC for the fulfillment of the degree.

EXAMINATIONS

The theory examination shall be three-hour duration for each paper at the end of every semester. The candidate failing in any subject(s) will be permitted to appear in the subsequent examination. The practical examinations for core subjects and SEC should be conducted at the end of every semester. Submission of record note books for practical examinations candidates appearing for practical examinations should submit bonafide Record note books prescribed for practical examinations, Otherwise the candidates will not be permitted to appear for the practical examinations. However, in genuine cases of the students who could not submit the record note books, they may be permitted to appear for the practical examinations, provided the concerned Head of the Department certify that the candidate has performed the experiments prescribed for the course. For such candidates zero (0) marks will be awarded for record note books.

REVISION OF REGULATIONS AND CURRICULUM

The University may revise/amend/ change the Regulations and Scheme of Examinations, as and when found necessary.

COMMENCEMENT OF THIS REGULATION:

This regulation shall take effect from the academic year 2026-2027, i.e., for students who are to be admitted to the first year of the programme during the academic year 2026-2027 and thereafter.

PATTERN OF QUESTION PAPER

Time: Three Hours
marks

Maximum: 75

Part A- (15X1= 15marks)

Answer all questions
(a),(b),(c),(d) Multiple Choice

Part B-(2X5=10 marks)
Answer any Two questions out of five (One question from each unit)

Part C-(5X10=50 marks) Answer
ALL questions
Either (or) Type Five questions (One question from each unit)

INTERNAL MARKS AND EXTERNAL MARKS

	Internal Marks	External Marks
Theory	25	75
Practical	40	60
Project	25	75
Internship/Health & Wellness	Commended/Highly Commended	

(a) Practical-External marks distribution (Total Marks: 60)

For each practical question the marks should be awarded as follows (External)

- i) Algorithm/flowchart -20%
- ii) Writing the program in the main answer book -30%
- iii) Test and debug the program -30%
- iv) Printing the correct output -20%

(Marks may be proportionately reduced for the errors committed in each of the above)

Practical Question Paper Pattern

Students should attend two questions (either or pattern)

COURSEOFSTUDY AND SCHEMEOFEXAINATION

SNO	Sem	Papercode	Part	Course Group	PaperType	SubjectTitle	Hours Week	ExamHours	Internal Marks	ExternalMarks	TotalMarks	
1	I		I	Foundation	Theory	Language–Tamil	6	3	25	75	100	
2			II	Foundation	Theory	Language–English	6	3	25	75	100	
3			III	Core	Theory	ProblemSolvingUsingC++	6	3	25	75	100	
4			III	Core	Practical	Practical:ProgramminginC++Lab	3	3	40	60	100	
5			III	Allied	Theory	Maths/Physics/Electronics/Statistics	6	3	25	75	100	
6			IV	NME-I	Theory		2	3	25	75	100	
7			IV	ValueAdded	Theory	ValueEducation	1	3	25	75	100	
Total					7		30					
8	II		I	Foundation	Theory	Language–Tamil	6	3	25	75	100	
9			II	Foundation	Theory	Language–English	6	3	25	75	100	
10			III	Core	Theory	DataStructuresandAlgorithms	5	3	25	75	100	
11			III	Core	Practical	Practical:DataStructuresusingC++Lab	3	3	40	60	100	
12			III	Allied	Theory	Maths/Physics/Electronics/Statistics	3	3	25	75	100	
13			III	Allied	Practical	Maths/Physics/Electronics/Statistics	2	3	40	60	100	
14			IV	NME-II	Theory		2	3	25	75	100	
15			IV	ValueAdded	Theory	DisasterManagement	1	3	25	75	100	
16					NaanMudhalvan	NMSD	2		25	75	100	
Total					7+1		30					2

Allied Subjects (Colleges can choose any one of the paper as Allied per year)

1.Mathematics 2.Physics 3.Statistics 4.Electronics

***NaanMudhalvan**–Courses–external 25 marks will be assessed by industry and internal will be offered by respective course teacher.

NaanMudhalvan reappear students–Alternate SKILL papers arranged

Non-Major Elective from 2026-27 onwards

Classes taken by Computer Science Department other majors

Semester	Paper Title
ODD	Fundamentals of Information Technology
EVEN	Office Automation

Problem Solving using C++	
Learning Objectives	
LO1	Understand the fundamentals of problem solving, algorithms, flowcharts, and object-oriented programming concepts.
LO2	Apply C++ programming constructs such as data types, operators, expressions, and control structures in developing programs.
LO3	Design and implement classes, objects, constructors, destructors, and member functions using object-oriented principles.
LO4	Demonstrate inheritance, operator overloading, polymorphism, and virtual functions in software development.
LO5	Develop C++ applications using pointers, file handling, and advanced object-oriented programming techniques.
CONTENT	
UNIT I: Introduction: Steps Involved in Problem Solving Using Computers- Characteristics of Algorithms- Symbols Used in Flow Charts - Pseudocode: Sequence, Selection and Iteration - Principles of Object-Oriented Programming: Basic Concepts of Object-Oriented Programming- Benefits of OOP- Applications of OOP - Beginning with C++: Simple C++ Program- Structure of C++ Program- Tokens, Expressions and Control Structures: Introduction- Tokens- Keywords- Identifiers and Constants- Basic Data Types- User Defined Data Types - Derived Data Types - Symbolic Constants	
UNIT II : Declaration of Variables - Operators in C++ - Scope Resolution Operator - TypeCast Operator - Expression and their Types - Assignment Expressions - Implicit Conversion - Operator Overloading - Operator Precedence - Control Structures- Function in C++ : Main Function - Function Prototyping - Call by Reference - Return by Reference - Inline Functions - Function Overloading - Friend and Virtual Functions - Math Library Functions.	
UNIT III: Classes and Objects: Specifying Class - Defining Member Functions - Nesting of Member Functions - Arrays within a Class - Memory Allocation for Objects - Static Data Members- Static Member Functions - Arrays of Objects - Object as Function Arguments - Friendly Functions - Returning Objects - const Member Functions - Constructors and Destructors: Constructors - Multiple Constructors in a Class - Constructors with Default Arguments - Dynamic Initialization of Objects - Copy Constructor - Dynamic Constructors - Destructors.	
UNIT IV: Operator Overloading and Type Conversion: Defining Operator Overloading - Overloading Unary Operators - Overloading Binary Operators - Rules for Overloading Operators- Type Conversions - Inheritance: Extending Classes: Defining Derived Classes - Single Inheritance - Multilevel Inheritance - Multiple Inheritance - Hierarchical Inheritance - Hybrid Inheritance - Virtual Base Classes - Abstract Classes.	
UNIT V: Pointers, Virtual Functions and Polymorphism: Introduction - Pointers- Pointers to Objects - Pointers to Derived Classes - Virtual Functions - Pure Virtual Functions - Working with Files: Classes for FileStream Operations- Opening and Closing a File- Detecting end-of-file- More about Open(): File Modes- Sequential Input and Output	

Operations-Updating a file: Random Access-Command-line Arguments.	
Course Outcomes	
CO1	Explain problem-solving techniques, algorithms, flowcharts, and the principles of object-oriented programming.
CO2	Develop C++ programs using operators, expressions, functions, and control structures.
CO3	Design programs using classes, objects, constructors, destructors, and member functions.
CO4	Apply inheritance, polymorphism, operator overloading, and type conversion concepts in application development.
CO5	Implement file handling, pointers, virtual functions, and dynamic binding concepts in C++ programs.
Learning Resources	
Text Books	1. "Problem Solving With C" Somashekara, M.T., Guru, D.S., Manjunatha, K.S. PHI Learning Pvt. Ltd., [UNIT-I-First Half] 2. Object Oriented Programming With C++, E. Balagurusamy Tata McGraw-Hill, New Delhi, 4 th Edition
Reference Books	1. ANSI and Turbo C++ by Ashoke N. Kamthane, Pearson Education 2. Programming: Principles and Practice Using C++, 3 rd Edition, Bjarne Stroustrup Addison-Wesley Professional Publisher. 3. C++: The Complete Reference-Schildt, McGraw-Hill Education (India)
Website/ Link	https://www.w3schools.com/cpp/cpp_oop.asp https://www.geeksforgeeks.org/cpp/object-oriented-programming-in-cpp/
Programming in C++ Lab	

Mapping with Programme Outcomes

CO/PSO	PSO1	PSO2	PSO3	PSO4	PSO5
CO1	3	2	1	1	1
CO 2	3	3	2	1	2
CO 3	2	3	3	2	2
CO 4	2	3	3	3	2
CO 5	2	2	3	3	3
Weightage of course contributed to each PSO	12	13	12	10	10

S-Strong-3M-Medium-2L-Low-1

Learning Objectives	
LO1	Understand the fundamentals of problem solving, algorithms, flowcharts, and object-oriented programming concepts.
LO2	Apply C++ programming constructs such as data types, operators, expressions, and control structures in developing programs.
LO3	Design and implement classes, objects, constructors, destructors, and member functions.
LO4	Demonstrate inheritance, operator overloading, polymorphism, and virtual functions in software development.
LO5	Develop C++ applications using pointers, file handling, and advanced object-oriented programming techniques.
LAB EXERCISES	
1. Write a C++ program to find the sum of the digits of a given number. 2. Write a C++ program to find the largest among three numbers using if-else. 3. Write a C++ program to make a simple calculator that can perform arithmetic operations. 4. Write a C++ program to find the factorial of a given number using recursion function. 5. Write a C++ program to demonstrate the usage of a constructor and destructor to manage the bank account records. 6. Write a C++ program to demonstrate polymorphism by calculating area of a rectangle and triangle using a shape class. 7. Write a C++ program using Binary operator (+, -, *, /) for Operator Overloading. 8. Write a C++ program using inheritance where Base class as "Student details" and Derived class as "Marks and Grade calculation". 9. Write a C++ program to reverse a given string using pointers. 10. Write a C++ program to count the number of characters in a file.	
Course Outcomes	
CO 1	Explain problem-solving techniques, algorithms, flowcharts, and the principles of object-oriented programming.
CO 2	Develop C++ programs using operators, expressions, functions, and control structures.
CO 3	Design programs using classes, objects, constructors, destructors, and member functions.
CO 4	Apply inheritance, polymorphism, operator overloading, and type conversion concepts in application development.
CO 5	Implement file handling, pointers, virtual functions, and dynamic binding concepts in C++ programs.

Mapping with Programme Outcomes

CO/PSO	PSO1	PSO2	PSO3	PSO4	PSO5
CO 1	3	2	1	1	1
CO 2	3	3	2	1	2
CO 3	2	3	3	2	2
CO 4	2	3	3	3	2
CO 5	2	2	3	3	3
Weightage of course contributed to each PSO	12	13	12	10	10

S-Strong-3M-Medium-2L-Low-1

Data Structures and Algorithms	
Learning Objectives	
LO1	Understand abstract data types and complexity analysis.
LO2	Develop problem-solving and logical thinking skills.
LO3	Design and implement efficient algorithms.
LO4	Apply DSA concepts using Python/C.
LO5	Apply DSA in real-world problem solving.
CONTENT	
UNIT I: Introduction and Overview: Definitions - Overview of Data Structures - Implementation of Data Structures - Abstract Data Types. Arrays: One Dimensional Array - Multidimensional Arrays.	
UNIT II: Linked Lists: Single Linked List - Double Linked List - Circular Linked List - Circular Double Linked List. Stacks: Stacks: Representation of Stack - Array Representation of Stacks - Operations on Stacks. Application of Stacks: Evaluation of Arithmetic Expressions - Implementation of Recursion - Factorial Calculation.	
UNIT III: Queues & Applications: Definition - Representation of Queues. Various Queue Structures: Circular Queue - Deque - Priority Queue. Application of Queues: Round Robin Algorithm. Trees: Trees: Basic Terminologies - Binary Trees. Operations on Binary Tree: Insertion - Deletion - Traversals. Types of Binary Trees: Expression Tree - Binary Search Tree - Heap Trees.	
UNIT IV: Graphs: Definitions - Graph Representations - Elementary Graph Operations: Depth First Search - Breadth First Search - Spanning Trees. Minimum Cost Spanning Trees: Kruskal's Algorithm - Prim's Algorithm. Sorting: Basic Terminologies - Selection Sort - Insertion Sort - Bubble Sort - Shell Sort - Quick Sort and Merge Sort.	
UNIT V: Searching: Linear Search Techniques - Linear Search with Array - Linear Search with Linked List. Non - Linear Search Techniques: Binary Tree Searching - Binary Search Tree Searching. Hashing: Introduction.	
Course Outcomes	
CO1	Understand abstract data types, Apply linear data structures such as arrays.
CO2	Apply linked lists to solve computational problems.
CO3	Apply linear data structures such as queues to solve computational problems. Design and implement tree.
CO4	Design and implement graph structures, and perform traversal techniques like BFS and DFS, Analyze the sorting
CO5	Apply searching, and hashing techniques to efficiently organize and retrieve

	data.
	Learning Resources
Text Books	1. Debasis Samanta "Classic Data Structures" PHI Learning Pvt Ltd, Second Edition, 2018. 2. Ellis Horowitz, Sarataj Sahni, Susan Anderson-Freed "Fundamentals of Data Structures in C" University Press, Second Edition, 2023.
Reference Books	1. A.V. Aho, J.D. Ullman, J.E. Hopcraft "Data Structures and Algorithms", Pearson Education. 2. Seymour Lipschutz, Data Structures McGraw Hill Publications, 2014, 1st Edition. 3. A Chitra, P T Rajan "Data Structures", Vijay Nicole Imprints Private Limited, 2016. 4. Reema Thareja, "Data Structures Using C" Oxford University Press, Third Edition 2023. 5. Samanta. D, Classic Data Structure Prentice Hall of India Pvt Ltd 2007, 9th Edition.
Website/ Link	1. https://www.geeksforgeeks.org/datastructures/ 2. https://www.tutorialspoint.com/data_structures_algorithms/index.htm 3. https://www.worldscientific.com/worldscibooks/10.1142/5256#t=aboutBook

CO/PSO	PSO1	PSO2	PSO3	PSO4	PSO5
CO 1	3	3	2	3	3
CO 2	3	2	3	2	2
CO 3	3	3	2	2	2
CO 4	3	2	3	3	2
CO 5	3	3	3	2	3
Weightage of course contributed to each PSO	15	13	13	12	12

S-Strong-3M-Medium-2L-Low-1

Mapping with Programme Outcomes

CO/PSO	PSO1	PSO2	PSO3	PSO4	PSO5
CO 1	3	2	1	1	1
CO 2	3	3	2	1	2
CO 3	2	3	3	2	2
CO 4	2	3	3	3	2
CO 5	2	2	3	3	3
Weightage of course contributed to each PSO	12	13	12	10	10

S-Strong-3M-Medium-2L-Low-1

DATA STRUCTURES USING C++ Lab	
Learning Objectives	
LO1	To understand the concepts of ADTs
LO2	To learn linear data structures - lists, stacks, queues
LO3	To learn Tree structures and application of trees
LO4	To learn graph structures and application of graphs
LO5	To understand various sorting and searching
LAB EXERCISES	
1. Write a C++ program to implement the LIST ADT using arrays. 2. Write a C++ program to multiply two matrices A and B and store the resultant matrix in C using arrays. 3. Write a C++ program to experiment the operation of STACK using array. 4. Write a C++ program to create menu driven options to implement QUEUE to perform the following (i) Insertion (ii) Deletion (iii) Modification (iv) Listing of elements 5. Write a C++ program to create Linked list representation of employee records and do the following operations using pointers. (i) To add a new record. (ii) To delete an existing record. (iii) To print the details about an employee. (iv) To find the number of employees in the structure. 6. Write a C++ program that uses stack operations to perform converting infix expression into postfix expression. 7. Write a C++ program to create a binary tree using Function to perform the Traversal of the binary tree in Preorder, inorder and postorder. 8. To write a C++ program to implement graph traversals by Breadth First Search and Depth First Search. 9. Write a C++ program for implementing the following searching methods i) Linear Search ii) Binary Search	

10. Write a C++ program that implements the following sorting methods to sort a given list of integers in ascending order

i) Bubble sort ii) Selection sort

Course Outcomes	
CO 1	Understand basic data structures such as arrays
CO 2	Analyze the use of pointers for linked list, doubly linked list
CO 3	Understand basic data structures such as stacks and queues
CO 4	Solve problems involving trees and graphs
CO 5	Apply algorithms for solving problems like sorting, searching, insertion and deletion of data.

Allied Subjects for B.Sc Electronics offered by the Department of Computer Science

Subject Title	SEMESTER I/III PAPER-I PROGRAMMING IN C	Semester	I/III
Subject Code	21UCSA05	Specialization	NA
Type	Allied: Theory	L:T:P:C	56:4:0:4

Course objective:

1. To apprehend the basic concepts of C-Programming language. This course introduces fundamental concepts such as arrays and structures.
2. It covers concepts such as arrays, pointers and file handling methods.
3. It provides technical skills to design and develop various applications.

CO Number	CO Statement	Knowledge Level
CO1	Recognize the Basic Terminologies of C Programming	K1
CO2	Understanding the statement structure and apply simple problems	K2, K3
CO3	Understand and apply the pre-defined functions and user-defined functions and then apply in simple problems	K3
CO4	Demonstrate the operation of Structures and unions.	K3, K4
CO5	Recognize the operation of Files	K3, K4

Subject Title	SEMESTER I/III PAPER-I PROGRAMMING IN C	Semester	I/III
Subject Code		Specialization	NA
Type	Allied: Theory	L:T:P:C	56:4:0:4
Unit	Contents	Levels	Sessions
I	Overview of C: History of C - Importance of C - Basic structure of C programs. Constants, variables and data types: Character set - C Tokens - Keywords and identifiers - Constants - Variables - Data types - Declaration of Variables- Declaration of storage classes - Assigning values to variables - Defining symbolic constants. Operators and expression: Types of Operators - Arithmetic Expressions- Evaluation of expressions - Precedence of arithmetic operators - Type conversions in expressions - Operator precedence and associativity. Managing input and output operations: Reading and writing a character - Formatted input and output.	K1	12
II	Decision making and branching: Simple IF, IF-ELSE, Nesting of IF-ELSE, ELSE-IF ladder, Switch statements- GOTO statements. Decision making and looping: WHILE statement - DO statement - FOR statement - Jumps in loops. Arrays: Definition & Detection - One dimensional - Two dimensional - Multi dimensional arrays - Dynamic arrays.	K2	12
III	Character arrays and strings: Introduction - Declaring and initializing string variables- Reading strings from terminal- Writing strings to screen - String handling functions - Table of strings. User - Defined functions: Introduction - Need for user - defined function - A Multi - function program - Elements of user - defined function - Definition of functions - Return values and their types - Function calls - Function declaration - All category of functions - Nesting of functions - Recursion - Passing arrays to functions - Passing strings to function.	K3	12
IV	Structures and Unions: Introduction-Defining a structure-Declaring structure variables-Accessing structure members-Structure initialization-Copying and comparing structure variables	K4	10

	- Arrays of structures - Arrays within structures -Structure within structures - Structures and functions - Unions - Size of structures - Bits fields.		
V	Pointers:Introduction-Understandingpointers- Accessingtheaddressofavariable- Initializingofpointervariables.Chainof pointers-Pointerexpression- Pointersandarrays-Pointersand characterstrings-Arraysofpointers- Pointersasfunctionarguments-Functionsreturningpointers- Pointerstofunctions- Pointerandstructures.FileManagement:Introduction-Defining and opening a file - Closing a file – Input/Output operation on files–ErrorhandlingduringI/Ooperations–Randomaccessfiles–Com mand linearguments.	K5	10
	LearningResources		
Textbooks	ProgramminginANSIC,E.BalgurusamyTataMcGrawHall,NewDelhi,5 th Edition.		
Reference Books	1. Schaum's outlines,programmingwithC,ByronSGottfried,2 nd Edition. 2. LetUsC.YashavantKanetkar.		
Website/ Link	http://www.learn-c.org/http://crasseux.com/books/ctutorial/		

MappingwithProgrammeOutcomes

CO Number	PO1	PO2	PO3	PO4
CO1	S	S	S	-
CO2	S	M	M	S
CO3	S	L	L	M
CO4	M	S	M	S
CO5	S	L	S	S

S-Strong,M-Medium,L-Low

Subject Title	PROGRAMMING IN VISUAL BASIC	Semester	II/IV
Subject Code		Specialization	NA
Type	Allied: Theory	L:T:P:C	56:4:0:4

Course objective:

- To introduce the basics of VB.
- To understand the concepts MDI Applications, ADO and ActiveX.
- To improve creative thinking in creating forms.

CO Number	CO Statement	Knowledge Level
CO1	Remember the basics of VB.	K1
CO2	Understand data and files in VB.	K2
CO3	Demonstrate the MDI Applications.	K3
CO4	Study of data control.	K4
CO5	Analyze the ADO and ActiveX.	K5

Subject Title	PROGRAMMING IN VISUAL BASIC	Semester	II / IV	
Subject Code		Specialization	NA	
Type	Allied: Theory	L:T:P:C	56:4:0:4	
Unit	Contents		Levels	Sessions
I	Welcome to Visual Basic – Creating an Application – IDE Forms and Controls – Variables in Visual Basic.		K1	10
II	Writing Code in Visual Basic – Working with File – Menu		K2	10
III	Multiple Document Interface Applications – Debugging Tips – The Common Dialog Control.		K3	12
IV	Introduction to Database – Working with the Data Control – Data Access Objects.		K4	12
V	ActiveX Data Objects – Crystal and Data Report – ActiveX.		K5	12
	Learning Resources			
Textbooks	Programming with Visual Basic 6.0, Mohammed Azam, Vikas Publishing House Pvt. Ltd., Chennai.			
Reference Books	1. Gary Cornell, "Visual Basic 6 from the Ground up", McGraw-Hill Education, 1998 2. Julia Case Bradley and Anita C. Millspaugh, "Programming in Visual Basic 6.0", Tata McGraw-Hill Edition, 2011.			
Website/ Link	• NPTEL & MOOC courses titled VB • https://www.freetutes.com/learn-vb6/			

Mapping with Programme Outcomes

CO Number	PS01	PS02	PS03	PS04
CO1	S	M	M	--
CO2	M	S	L	-
CO3	S	M	L	M
CO4	S	M	M	L
CO5	S	M	L	L

S-Strong, M-Medium, L-Low

Subject Title	PROGRAMMING IN C & VISUAL BASIC PRACTICAL	Semester	II/IV
Subject Code		Specialization	NA
Type	Allied: Practical	L:T:P:C	30:0:2:2

COURSE OBJECTIVE:

1. To impart Practical Training in C Programming Language.
2. Familiarize the different control and decision making statements in—C++.
3. Build programs using arrays and strings.
4. Provide knowledge on working with files and functions.

PROGRAMMING IN C PRACTICAL LIST:

1. Create a program to find the Simple Interest.
2. Create a program to find the Arithmetic Mean and Standard Deviation.
3. Create a program to find the Biggest value among given 3 number.
4. Create a program to calculate the Area of perimeter of square and rectangle.
5. Create a program to convert Binary to Decimal conversion.
6. Create a program to convert Decimal to Binary conversion.
7. Create a program to print the Fibonacci series using Recursion.
8. Create a program to swap the given two integers.
9. Create a program to print the factorial of a number.
10. Create a program to display the multiplication table.

PROGRAMMING IN VISUAL BASIC PRACTICAL LIST:

1. Write a VB program to implement Forms.
2. Write a VB program to implement Inputbox, and MessageBox.
3. Write a VB program to implement Control Statements and Loops.

4. Write a VB program to implement Commandbox, Option button, and Checkbox.
5. Write a VB program to implement Combobox, Listbox, and Scrollbars.
6. Write a VB program to implement Timer.
7. Write a VB program to implement MDI Forms.
8. Write a VB program to implement DAO.
9. Write a VB program to implement ADO.
10. Write a VB program to implement a Calculator.

COURSE OUTCOME:

1. Study all the Basic Statements in C Programming.
2. Practice the usage of branching and looping statements.
3. Apply string functions and arrays usage.
4. Analyse the use of pointers and files.
5. Understand the features in VB.
6. Select and apply statements for design forms.
7. Combine multiple features in interface and database.

Title of the Course		WEBDESIGNINGWITHHTML (ForB.ScMATHEMATICSWITHCOMPUTERAPPLICATION)					
PaperNumber		ELECTIVECOURSEI					
Category	Elective	Year	I	Credits	3	Course Code	
		Semester	I				
InstructionalHours perweek		Lecture	Tutorial		LabPractice	Total	
		3	-		1	4	
Pre-requisite		12 th StandardMathematics					
Objectives of the Course		• Insertagrophicwithinawebsite. • Createalinkwithinawebsite. • Createatablewithinawebsite. • Inserttheadinglevelswithinawebsite. • Insertorderedandunorderedlistswithinawebsite.Createawebsite.					
CourseOutline		UNIT I-Introduction to HTML – Opening for writing HTML – Unicode Transformation Format – HTML 5 Resources – What is different in HTML 5? - <DOCTYPE> in HTML 5					
		UNITII- DesigningaWebpage:DesignConsiderationsandPlanning–BasicTag sandDocumentstructure–HTMLTags <HTML>...</HTML>-HeadTags<HEAD>...</HEAD>- TitleTags–BodyTags<BODY>...</BODY>-Metadata–Savingan HTML document – Actions.					
		UNITIII- Formatting: PageFormatting–AddingaNew Paragraph–Adding a Line Break – Inserting Blank Space – Preformatted Text – Changing a Page’s Background Color – Div Element - Text items and objects – Headings – Comments – Block Quotes – Horizontal Lines – Special Characters – Creating Lists – Numbered (Ordered) Lists – Bulleted(Unordered)Lists–Nested Lists-DefinitionLists.					
		UNIT IV-Links: Introduction to Links – Text Links – Image Links – Opening awebpagein anewwindow/Tab–Setting AllLinkson a page to open in a new window/Tab – Linking to an area on thesame page (Bookmarks)– Linking to an E-mail Address– Linking toothertypesofFiles.					
		UNIT V- Images: Introduction to Images: Adding Images –Resizing images – Alternative (ALT) Text – Image Labels. Tables: Introduction toTables-InsertingaTable–TableBorders-Table Headers					

Practical Course Outline	1. Write a program to illustrate the basic tags of HTML. 2. Write a program on Page formatting. 3. Write a program to illustrate paragraph tag. 4. Write a program to change background colour. 5. Write a program to create a list (Numbered (Ordered) Lists – Bulleted (Unordered) Lists). 6. To create a HTML file using special characters. 7. To create a HTML file containing a hyperlink. 8. Write a HTML program to display a table with 5 rows and 4 columns with appropriate heading. 9. Write a HTML code to design a complex nested list. 10. Write a HTML code to develop a web page having two frames that divide the page into two equal rows and divide the first row into two columns.
Skills acquired from this course	Learn the language of the web: HTML. Understand the principles of creating an effective web page. Learn to embed other media links into web pages.
Recommended Text	1. —Mastering HTML5 and CSS3 Made Easy II, Teach U Comp Inc., 2014. 2. Thomas Michaud, —Foundations of Web Design: Introduction to HTML & CSS II
Website and e-Learning Source	1. https://www.teachucomp.com/samples/html/5/manuals/Mastering-HTML5-CSS3.pdf 2. https://www.w3schools.com/html/default.asp

METHOD OF EVALUATION:

Continuous Internal Assessment	End Semester Examination		Total
	Theory	Practical	
25	50	25	100

Course Learning Outcomes (for Mapping with POs and PSOs)

Students will be able to

CLO1: Understand the basic concept in HTML. Concept of resources in HTML

CLO2: Create the Meta Data, Design concept & save the files.

CLO3: Understand page formatting and the concept of list.

CLO4: Creating Links and understand the concept of creating link to email

address **CLO5:** Create concepts by adding images. Understand the table creation.

	POs						PSOs		
	1	2	3	4	5	6	1	2	3
CLO1	3	2	1	-	3	2	2	2	2
CLO2	3	2	1	-	3	2	2	2	2
CLO3	3	2	1	1	3	2	2	2	2
CLO4	3	2	1	-	3	2	2	2	2
CLO5	3	2	1	-	3	2	2	2	2

3-StrongCorrelation

2-MediumCorrelation

1-LowCorrelation

TitleoftheCourse		PROGRAMMINGWITHPYTHON (FORB.ScMATHEMATICSWITHCOMPUTERAPPLICATION)					
PaperNumber		ELECTIVEPAPERII					
Category	Elective	Year	I	Credits	3	Course Code	
		Semester	II				
Instructional Hours perweek		Lecture		Tutorial	LabPractice	Total	
		3		--	1	4	
Pre-requisite		12 th StandardMathematics					
Objectivesofthe Course		• DescribethecoresyntaxandsemanticsofPython programming language. • Discovertheneedforworkingwiththestringsandfunctions. • Illustratetheprocessofstructuringthedatausinglists, dictionaries, tuples and sets. • UnderstandtheusageofpackagesandDictionaries • Toknowthecostsandprofitmaximization					
CourseOutline		UNITI -IntroductiontoPython–Origins–Features–Downloadingand InstallingPython–RunningPython–PythonDocumentation.Getting Started–ProgramOutputstatement–ProgramInputfunction– PythonBasics–Statementsandsyntax–VariableAssignment– Identifiers – Numbers – Introduction – Integers – Double Precision Floating Point Numbers – Complex Numbers – Operators – Built-in functionsforallnumericatypes.					
		UNITII - Sequences:Strings,ListsandTuples–Sequences–Strings–Strings and Operators–String-Only Operators–Built-in Functions– StringBuilt-inMethod–Lists–Operators-Built-inFunctions–ListType Built-inMethods–Tuples—TupleOperatorsandBuilt-inFunctions-					

	UNIT III–Conditionals and Loops–If statement–else statement–elif statement–Conditional expressions–while statement–for statement–break statement–Continue statement–pass statement –Functions and Functional Programming–Calling Functions–Creating Functions– Passing Functions–Formal Arguments–Variable–Length Arguments. UNIT IV–Errors and Exceptions –Exceptions in Python –Detecting and Handling Exceptions Context Management – with statement – Raising Exceptions – Modules – Modules and Files – Name spaces – Importing Modules – Features of Module - Import –Module Built-in Functions–Packages. UNIT V–Files and Input / Output: File Objects – File Built-in Functions – File Built-in Methods – File Built-in Attributes – Command-Line Arguments - File System –Object-oriented Programming–Introduction–Classes–Class Attributes–Instances–Instance Attributes.
Practical Course Outline	1. Program for System configuration 2. Working with Strings 3. Working with Lists 4. Working with Tuples 5. Working with Dictionary 6. Working with conditional loops–if, else, elif 7. Working with conditional expressions–for, while, break, continue 8. Implementing programs on functions 9. Working with function–formal arguments and variable-length arguments 10. Working with Detecting and Handling Exception 11. Working with modules 12. Working with Built-in Functions
Skills acquired from this course	1. Impart knowledge and skill in getting started with Python basic concepts. 2. Expose to the concepts of sequences, string and built-in function of python. 3. Introduce the various control statements and looping for decision making. 4. Study the exceptions and error handling in program execution. 5. Gain knowledge on file management in Python Programming.
Recommended Texts	Wesley J. Chun, —Core Python Programming II, 2 nd Edition, Pearson Education LPE, New Delhi, 2007.

ReferenceBooks	1. Mark Summerfield, Programming in Python 3, Pearson Education LPE, New Delhi, 1996. 2. Python Programming, Braindraper, kindle unlimited pvt. ltd. 3. Core Python Programming, Dr. R. Nageswara Rao, dreamtech pvt ltd. Kindle. 4. The complete reference on Python, Martin. C. Brown MAC Graw Hill pvt. ltd. 5. Coding for beginners using Python. Louie Stowell, kindle publishing pvt. ltd.
Website and e-Learning Source	1. https://www.programiz.com/python-programming 2. https://www.guru99.com/python-tutorials.html 3. https://www.w3schools.com/python/python_intro.asp 4. https://www.geeksforgeeks.org/python-programming-language/ 5. https://en.wikipedia.org/wiki/Python_(programming_language)

METHOD OF EVALUATION:

Continuous Internal Assessment	End Semester Examination		Total
	Theory	Practical	
25	50	25	100

Course Learning Outcome (for Mapping with POs and PSOs)

Students will be able to

CLO1: Develop and execute simple Python programs.

CLO2: Write simple Python programs using conditionals and looping for solving problems.

CLO3: Decompose a Python program into functions.

CLO4: Represent compound data using Python lists, tuples, dictionaries etc.

CLO5: Read and write data from/to files in Python programs.

	POs						PSOs		
	1	2	3	4	5	6	1	2	3
CLO1	3	2	1	1	3	2	2	2	2
CLO2	3	2	1	1	3	2	2	2	2
CLO3	3	2	1	1	3	2	2	2	2
CLO4	3	2	1	1	3	2	2	2	2
CLO5	3	2	1	1	3	2	2	2	2

3-StrongCorrelation 2-MediumCorrelation1-LowCorrelation

Title of the Course		PAPER I- C PROGRAMMING LANGUAGE AND PRACTICAL (FOR B.Sc MATHEMATICS)						
Paper Number		C PROGRAMMING LANGUAGE						
Category	Core	Year	I	Credits	5	Course Code		
		Semester	I					
Instructional Hours per week		Lecture		Tutorial		Lab Practice		Total
		4		-		2		6
Pre-requisite		12 Th Standard Mathematics						
Objectives of the Course		• It is the study of programming language • Study about constants, variables and data types • Study about operators and Expressions • Study of Managing Input and Output Operations						
Course Outline		UNIT-I: Constants, Variables and Data Types: Characteristic Set – C Tokens – Keywords and Identifiers – Constants – Variables. (Chapter 2: Section 2.1 to 2.6).						
		UNIT-II: Constants, Variables and Data Types: Data Types – Declaration of Variables – Declaration of Storage Class – Assigning Values to Variables – Defining Symbolic Constants. (Chapter 2: Section 2.7 to 2.11).						
		UNIT-III: Operations and Expressions: Arithmetic Operators – Relation Operators – Logical Operators – Assignment Operators – Increment and Decrement Operators – Conditional Operators – Bitwise Operators – Special Operators. (Chapter 3: Sections 3.2 to 3.9).						
		UNIT-IV: Operations and Expressions: Arithmetic Expressions – Evaluation of Expression – Precedence of Arithmetic Operators – Some Computational Problems – Type Conversions in Expressions. (Chapter 3: Sections 3.10 to 3.14)						
		UNIT-V: Managing Input and Output Operations: Reading a Character – Writing a Character – Formatted Input – Formatted Output. (Chapter 4: Sections 4.2 to 4.5)						
Skills acquired from this course		Knowledge, Analytical ability.						

RecommendedText	1.E.Balagurusamy–ProgramminginANSIC,FifthEdition,TataMcGraw Hill Education Private Limited, New Delhi.
ReferenceBooks	1. C. Xavier - C. Language and Numerical Methods, Years ofPublication 1999, New age international limited, New Delhi. 2 Kernighan B.W. and RatchineD.M.–TheCProgrammingLanguage,PrenticeHallIndia,NewDelhi 1997.
Websiteand e-LearningSource	https://nptel.ac.in

CourseOutcomes(COs)

Onsuccessfulcompletionofthecourse,thestudentwillbeableto

CO Number	COStatement
CO1	DefineConstantsandvariables.
CO2	DefineDataTypesandexamples
CO3	DefineOperatorsandexamples
CO4	DefineExpressionsandexamples
CO5	DefineInputandoutputOperations

MappingofCOswithPOs

PO CO	PO1	PO2	PO3	PO4	PO5
CO1	3	2	2	3	3
CO2	2	3	3	3	3
CO3	3	3	3	3	3
CO4	2	3	3	2	3
CO5	2	3	3	3	3

Title of the Course		PAPER II - C PROGRAMMING LANGUAGE AND PRACTICAL (FOR B.Sc MATHEMATICS)					
Paper Number		C PROGRAMMING LANGUAGE					
Category	Core	Year	I	Credits	3	Course Code	
		Semester	II				
Instructional Hours per week		Lecture		Tutorial		Lab Practice	Total
		4		-		2	6
Pre-requisite		12 Th Standard Mathematics					
Objectives of the Course		• It is the study of programming language • Study about Decision making and Branching • Study about Decision making and Looping • Study about Character arrays and Stings • Study about Use-defined functions					
Course Outline		UNIT-I: Decision making and Branching: Decision Making with IF Statement – Simple IF Statement – The IF ... ELSE Statement – Nesting of IF ... ELSE Statement – The ELSE IF Ladder – The Switch Statement. (Chapter 2: Section 5.2 to 5.7).					
		UNIT-II: Decision making and Looping: The WHILE Statement – The DO Statement – The FOR Statement – Jumps in LOOPS (Chapter 6: Section 6.2 to 6.5).					
		UNIT-III: Arrays: One Dimensional Arrays – Declaration of One Dimensional Arrays – Initialization of One dimensional Arrays – Two Dimensional Arrays – Initializing Two dimensional Arrays – Multi Dimensional Arrays. (Chapter 7: Sections 7.2 to 7.7).					
		UNIT-IV: Character Arrays and Strings: Declaring and Initializing String Variable – Reading Strings from Terminal – Writing Strings to Screen – Arithmetic Operations on Characters. Chapter 8: Sections 8.2 to 8.5)					
		UNIT-V: User – defined Functions: Need for User-defined Functions – A multi-function Program – Elements of User-defined Functions – Definition of functions – Return Values and their Types. (Chapter 9: Sections 9.2 to 9.6).					

Skills acquired from this course	Knowledge, Analytical ability.
Recommended Text	1. E. Balagurusamy – Programming in ANSI C, Fifth Edition, Tata McGraw Hill Education Private Limited, New Delhi.
Reference Books	1. C. Xavier - C Language and Numerical Methods, Year of Publication 1999, New age international limited, New Delhi. 2. Kernighan B. W. and Ritchie D. M. – The C Programming Language, Prentice Hall India, New Delhi 1997.
Website and e-Learning Source	https://nptel.ac.in

Course Outcomes (COs)

On successful completion of the course, the students will be able to

CO Number	CO Statement
CO1	Define Decision making and Branching
CO2	Define Decision making and looping
CO3	Define Arrays and examples
CO4	Define Character Arrays and Strings
CO5	Define User-defined Functions

Mapping of COs with POs

PO \ CO	PO1	PO2	PO3	PO4	PO5
CO1	3	2	2	3	3
CO2	2	3	3	3	3
CO3	3	3	3	3	3
CO4	2	3	3	2	3
CO5	2	3	3	3	3

Non Major Elective from 2026-27 onwards

Subject Name	Category	L	T	P	S	Inst. hours	Credits	Marks		
								CIA	Exter	Total
Fundamentals of Information Technology	Non Major Elective Odd Sem	2	-	-	-	2	2	25	75	100
Understand basic concepts and terminology of information technology.										
Have a basic understanding of personal computers and their operation										
Be able to identify data storage and its usage										
Get great knowledge of software and its functionalities										
Understand about operating system and their uses										
Contents										
UNIT - I: Introduction to Computers										
Introduction, Definition, Characteristics of computer, Evolution of Computer, Block Diagram of a computer, Generations of Computer, Classification of Computers, Applications of Computer, Capabilities and limitations of computer.										
UNIT - II: Basic Computer Organization										
Role of I/O devices in a computer system. Input Units: Keyboard, Terminals and its types. Pointing Devices, Scanners and its types, Voice Recognition Systems, Vision Input System, Touch Screen. Output Units: Monitors and its types. Printers: Impact Printers and its types. Non-Impact Printers and its types, Plotters, types of plotters, Sound cards, Speakers.										
UNIT - III: Storage Fundamentals										
Primary Vs Secondary Storage, Data storage & retrieval methods. Primary Storage: RAM, ROM, PROM, EPROM, EEPROM. Secondary Storage: Magnetic Tapes, Magnetic Disks, Cartridge tape, hard disks, Floppy disks, Optical Disks, Compact Disks, Zip Drive, Flash Drives.										
UNIT - IV: Software										
Software and its needs, Types of S/W. System Software: Operating System, Utility Programs. Programming Language: Machine Language, Assembly Language, High Level Language their advantages & disadvantages. Application S/W and its types: Word Processing, Spreadsheets, Presentation, Graphics, DBMS s/w.										
UNIT - V: Operating System										
Functions, Measuring System Performance, Assemblers, Compilers and Interpreters. Batch Processing, Multiprogramming, Multi-Tasking, Multiprocessing, Time Sharing, DOS, Windows, Unix/Linux.										

	Course Outcomes	Programme Outcomes
CO	On completion of this course, students will:	On completion of this course, students will:
CO1	Learn the basics of computer, Construct the structure of the required things in computer, learn how to use it.	PO1, PO2, PO3, PO4, PO5, PO6
CO2	Develop organizational structure using for the devices present currently under input or output unit.	PO1, PO2, PO3, PO4, PO5, PO6
CO3	Concept of storing data in computer using two headers namely RAM and ROM with different types of ROM with advancement in storage basis.	PO1, PO2, PO3, PO4, PO5, PO6
CO4	Work with different software, Write program in the software and applications of software.	PO1, PO2, PO3, PO4, PO5, PO6
CO5	Usage of Operating system in information technology which really acts as an interpreter between software and hardware.	PO1, PO2, PO3, PO4, PO5, PO6

Textbooks

1	Anoop Mathew, S. Kavitha Murugesan (2009), — Fundamentals of Information Technology II, Majestic Books.
2	Alexis Leon, Mathews Leon, II Fundamentals of Information Technology II, 2 nd Edition.
3	S.K Bansal, — Fundamentals of Information Technology II.

Reference Books

1.	Bhardwaj Sushil Puneet Kumar, — Fundamentals of Information Technology II
2.	GG WILKINSON, — Fundamentals of Information Technology II, Wiley-Blackwell
3.	A Ravichandran, — Fundamentals of Information Technology II, Khanna Book Publishing

Web Resources

1.	https://testbook.com/learn/computer-fundamentals
2.	https://www.tutorialsmate.com/2020/04/computer-fundamentals-tutorial.html
3.	https://www.javatpoint.com/computer-fundamentals-tutorial
4.	https://www.tutorialspoint.com/computer_fundamentals/index.htm
5.	https://www.nios.ac.in/media/documents/sec229new/Lesson1.pdf

Mapping with Programme Outcomes:

CO/PSO	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	3	3	3	3	3	3

CO2	3	3	3	3	3	3
CO3	3	3	3	3	3	3
CO4	3	3	3	3	2	3
CO5	3	3	2	3	3	2
Weightageofcourse contributedtoeach PSO	15	15	14	15	14	14

S-Strong-3 M-Medium-2 L-Low-1

SubjectCode	SubjectName	Category	L	T	P	S	C	I	Marks		
	OFFICEAUTOMATION	Non Major Elective Even Sem							CIA	External	Total
			2	-	-	-	2	2			
Learning Objectives											
LO1	Understand the basics of computer systems and its components.										
LO2	Understand and apply the basic concepts of a word processing package.										
LO3	Understand and apply the basic concepts of electronic spreadsheet software.										
LO4	Understand and apply the basic concepts of data base management system.										
LO5	Understand and create a presentation using Power Point tool.										
UNIT	Contents									No.of Hours	
I	Introductory concepts: Memory unit– CPU-Input Devices: Keyboard, Mouse and Scanner. Output devices: Monitor, Printer. Introduction to Operating systems & its features: DOS– UNIX–Windows. Introduction to Programming Languages.									6	
II	WordProcessing:Open,Saveandcloseworddocument;Editingtext–tools,formatting,bullets;SpellChecker-Documentformatting – Paragraph alignment, indentation, headers and footers, numbering; printing–Preview, options, merge.									6	
III	Spreadsheets:Excel–opening,enteringtextanddata,formatting,navigating;Formulas–entering,handlingand copying; Charts–creating, formatting and printing,analysistables,preparationoffinancialstatements,introductionto dataanalytics.									6	
IV	Database Concepts: The concept of data base management system; Data field, records, and files, Sorting and indexing data; Searching records. Designing queries, and reports; Linking of datafiles; Understanding Programming environment in DBMS; Developing menu drive applications in query language (MS–Access).									6	
V	Power point: Introduction to Power point - Features–Understanding slide typecasting &viewing slides – creating slide shows. Applying special object – including objects & pictures – Slide transition–Animation effects, audio inclusion, timers.									6	
	Total									30	

Course Outcomes		ProgrammeOutcomes
CO	Oncompletionofthiscourse,studentswill	
CO1	Possesstheknowledgeonthebasicsofcomputersandits components	PO1,PO2,PO3,PO6,PO8
CO2	GainknowledgeonCreatingDocuments,spreadsheetandp resentation.	PO1,PO2,PO3,PO6
CO3	LearntheconceptsofDatabaseandimplementtheQueryin Database.	PO3,PO5,PO7
CO4	Demonstratetheunderstandingof different automationtools.	PO3,PO4,PO5,PO7
CO5	Utilizetheautomationtoolsfordocumentation, calculationandpresentationpurpose.	PO4,PO6,PO7,PO8
TextBook		
1	PeterNorton,—IntroductiontoComputersII–TataMcGraw-Hill.	
ReferenceBooks		
1.	JenniferAckermanKettel,GuyHat-Davis,CurtSimmons,—Microsoft2003II,Tata McGrawHill.	
WebResources		
1.	https://www.udemy.com/course/office-automation-certificate-course/	
2.	https://www.javatpoint.com/automation-tools	

Mapping with Programme Outcomes:

MAPPING TABLE						
CO/PSO	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	3	2	2	3	3	3
CO2	3	3	3	3	3	3
CO3	3	3	3	3	3	3
CO4	3	3	3	3	3	3
CO5	3	3	3	3	3	3
Weightage of course contributed to each PSO	15	14	14	15	15	15

S-Strong-3 M-Medium-2 L-Low-1