

DHANRAJ BAID JAIN COLLEGE

(Autonomous)

Co-Educational Minority Institution
Owned & Managed by Tamilnadu Educational and Medical Trust
Approved by Government of Tamil Nadu & Affiliated to University of Madras
Re-Accredited by NAAC with "A" Grade
Thoraipakkam, Chennai – 600 097.

DEPARTMENT OF COMPUTER SCIENCE

B.Sc (Computer Science)



SYLLABUS

(Choice Based Credit System)

(Effective from the Academic year 2026-2027)

Total No. of Semesters: 6

Total No. of Credits: 141

SCIENCE PROGRAM OUTCOMES

- PO1. Science:** Apply the knowledge of science fundamentals to solve problems in chosen field.
- PO2. Problem Analysis:** Identify and review literature, thereby analyzing problems to arrive at substantiated solutions using the basic principles of science.
- PO3. Science Graduate and Society:** Apply reasoning to assess the societal issues like health, safety, legal and cultural to dominate the need for sustainable development.
- PO4. Ethics:** Apply ethical principles and commit to the professional ethics and norms.
- PO5. Environment and Sustainability:** To understand the impact of responsibility of pursuing the environment and demonstrate the need for sustainable development.
- PO6. Individual and Team Work:** Function effectively as an individual and as member or leader in diverse teams in multi – disciplinary settings.
- PO7. Communication:** Speak, read, write and listen clearly in person and through electronic media in English and in one Indian language, and make meaning of the world by connecting people, ideas, books, media and technology.
- PO8. Project Management and Finance:** Demonstrate the knowledge of the understanding of the fundamental principle of managing a project and apply the same in one 's own work as a member and as a leader of a team, to manage project in a multi-disciplinary environment.
- PO9. Life – Long Learning:** Recognize the need for and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change

PROGRAM SPECIFIC OUTCOME

PSO1: Think in a critical and logical based manner

PSO2: Familiarize the students with suitable software tools of computer science and industrial applications to handle issues and solve problems in mathematics or statistics and real time application related sciences.

PSO3: Understand, formulate, develop programming model with logical approaches to a Address issues arising in social science, business and other contexts.

PSO4: Acquire good knowledge and understanding to solve specific theoretical and applied problems in advanced areas of Computer science and mathematics.

PSO5: Provide students/learners sufficient knowledge and skills enabling them to undertake further studies in Computer Science or Applications or Information Technology and its allied areas on multiple disciplines linked with Computer Science.

PSO6: Equip with Computer science technical ability, problem solving skills, creative talent and power of communication necessary for various forms of employment, internships& societal activities.

**BACHELOR DEGREE COURSE
B.Sc. - COMPUTER SCIENCE
SEMESTER SYSTEM WITH CREDITS
(Effective from the academic year 2026-2027)**

REGULATIONS

1. ELIGIBILITY FOR ADMISSION

Candidates for admission to the first year of the Degree of Bachelor of Science (Computer Science) Courses shall have passed the Higher Secondary Examinations, having **Mathematics** as one of the subject, conducted by the Government of Tamil Nadu or an Examination accepted as equivalent thereof by the Syndicate of the University of Madras.

2. ELIGIBILITY FOR THE AWARD OF DEGREE

A Candidate shall be eligible for the award of the degree only if he/she has undergone the prescribed course of study in our College which is affiliated to the University of Madras for a period of not less than three academic years, passed the examinations of all the Six Semesters prescribed earning 140 credits and fulfilled such conditions as have been prescribed therefor.

3. DURATION

- a) Each academic year shall be divided into two semesters. The first academic year shall comprise the first and second semesters, the second academic year comprise of the third and fourth semesters and the third academic year shall comprise the fifth and sixth semesters respectively.
- b) The odd semesters shall consist of the period from June to November of each year and the even semesters from December to April of each year. There shall be not less than 90 working days for each semester.

4. COURSE OF STUDY

The Main Subject of study for Bachelor Degree shall consist of the following

PART-I: Tamil or any other Modern (Indian or Foreign) or Classical Languages.

PART-II: English

PART-III: CORE COURSES Consisting of (a) Core Course (b) Elective Course (Generic / Discipline Specific)

PART-IV: Skill Enhancement Course (SEC), Foundation Course (FC), Environmental Studies, Value Education, Internship/ Industrial Training, Professional Competency Skill Enhancement Course.

PART- V: Extension Activity

5. SCHEME OF EXAMINATION

Scheme of Examination shall be enclosed in APPENDIX - I

6. REQUIREMENTS FOR PROCEEDING TO SUBSEQUENT SEMESTER

- I. Candidates shall register their names for the First Semester Examination after the admission in the UG Courses.
- II. Candidates shall be permitted to proceed from the First Semester up to final semester irrespective of their failure in any of the semester examinations subject to the condition that the candidates should register for all the arrear subjects of earlier semesters along with current (subsequent) Semester Subjects.
- III. Candidates shall be eligible to go to subsequent semester, only if they earn, sufficient attendance as prescribed therefore by the syndicate from time to time, provided in case of a candidate earning less than 50% of attendance in any one of the Semesters due to any extraordinary circumstance such as medical grounds, such candidates who shall produce Medical Certificate issued by the Authorised Medical Attendant (AMA), duly certified by the Principal of the College, shall be permitted to proceed to the next semester and to complete the Course of Study. Such candidates shall have to repeat the missed semester by rejoining after completion of Final Semester of the course, after paying the fee for the break of study as prescribed by the University from time to time.

7. PASSING MINIMUM

A candidate shall be declared to have passed in each paper/practical of the Main Subject of study wherever prescribed, if he/she secures NOT LESS THAN 40% of the marks prescribed for the examination. He/she shall be declared to have passed the whole examination, if he/she passes in all the papers and practicals wherever prescribed as per the scheme of examinations earning 140 CREDITS. He/she shall also fulfill the compulsory extension services prescribed earning a minimum of 1 credit out of maximum of 2 credits to qualify for the degree.

8. CLASSIFICATION OF SUCCESSFUL CANDIDATES

- i) PART I: LANGUAGE OTHER THAN ENGLISH: Successful candidates passing the examinations for the Language and securing the marks (1) 60 percent and above and (2) 50 percent and above but below 60 percent in the aggregate shall be declared to have passed the examination in the FIRST and SECOND Class, respectively. All other successful candidates shall be declared to have passed the examination in the THIRD Class.
- ii) PART II: ENGLISH: Successful candidates passing the examinations for English and securing the marks (1) 60 percent and above and (2) 50 percent and above but below 60 percent in the aggregate shall be declared to have passed the examination in the FIRST and SECOND class, respectively. All other successful candidates shall be declared to have passed the examination in the THIRD class.
- iii) PART III: CORE COURSE: Consisting of (a) Core Course (b) Elective Course (Generic / Discipline Specific)

Successful candidates passing the examinations for Core Courses together and securing the marks 1)60 percent and above (2) 50 percent and above but below 60 percent in the aggregate of the marks prescribed for the Core Courses together shall be declared to have passed the examination in the FIRST and SECOND class respectively. All other successful candidates shall be declared to have passed the examinations in the THIRD class.

iv) PART IV: Skill Enhancement Course (SEC), Foundation Course (FC), Environmental Studies, Value Education, Internship/ Industrial Training, Professional Competency Skill Enhancement Course.

v) PART V: COMPULSORY EXTENSION SERVICE:

Successful Candidates earning a minimum of 1 credit or a maximum of 2 credits SHALL NOT BE taken into consideration for Classification/Ranking/Distinction.

9. RANKING

Candidates who pass all the examinations prescribed for the course in the FIRST APPEARANCE ITSELF ALONE are eligible for Classification/Ranking/Distinction.

Provided in the case of Candidates who pass all the examinations prescribed for the course with a break in the First Appearance due to the reasons as furnished in the Regulations.7.

(iii) supra are only eligible for Classification / Distinction.

PATTERN OF QUESTION PAPER

Maximum Marks: - 100 Marks

Time: - 3 Hours

Part – A (50 Words)

To answer any TEN Questions out of Twelve Questions

10x2=20 Marks

Part – B (200 Words)

To answer any FIVE Questions out of Eight Questions

5 x 7=35 Marks

Part – C (500 Words)

To answer Any THREE Questions out of Five Questions

3x15=45 Marks

QUESTION PAPER FOR PRACTICALS

The external examiner will prepare a question paper on the Spot with the help of the Question Bank Supplied by the Controller's Office.

B.Sc. Computer Science

First Year

Semester-I

Part	List of Courses	Credit	Hours per week (L/T/P)
Part-I	Language	3	5
Part-II	English	3	5
Part-III	CC1-Object Oriented Programming with C++	5	5
	CC2- Practical – Object Oriented Programming Lab	5	5
	Elective Course - (Generic / Discipline Specific) - EC1 - Statistics	3	5
Part-IV	Skill Enhancement Course SEC-1 (Non Major Elective)- Office Automation Lab	2	3
	Foundation Course FC- Problem Solving Techniques	2	2
		23	30

Semester-II

Part	List of Courses	Credit	Hours per week (L/T/P)
Part-I	Language	3	5
Part-II	English	3	5
Part-III	CC3-Python Programming	5	5
	CC4-Python Programming Lab	5	5
	Elective Course - (Generic / Discipline Specific) - EC2 – Mathematics	3	5
Part-IV	Skill Enhancement Course -SEC-2 (Non Major Elective) – Understanding Internet Technology	2	2
	Skill Enhancement Course -SEC-3 (Discipline Specific / Generic)- AI Automation Lab	2	3
		23	30

First Year (Semester I)

Title of the Course/ Paper	Subject Name	Category	L	T	P	S	Credits	Inst. Hours	Marks		
									CIA	External	Total
Core Course I	Object Oriented Programming with c++	Core	Y	-	-	-	5	5	40	60	100
Course Objective											
CO1	Explain the fundamentals of C++, Object-Oriented Programming concepts, control structures, and functions to develop simple C++ programs.										
CO2	Develop C++ programs using classes, objects, constructors, destructors, static members, friend functions, and arrays of objects.										
CO3	Apply operator overloading and inheritance concepts to implement reusable and efficient object-oriented programs.										
CO4	Demonstrate the use of pointers, dynamic memory allocation, arrays of objects, and polymorphism for solving programming problems.										
CO5	Develop C++ applications using file handling, templates, exception handling, and string manipulation techniques.										
UNIT	Details										No. of Hours
I	Introduction to C++ – Features and Advantages of C++ – Basics of Object-Oriented Programming – Structure of a C++ Program – Variables, Constants and Data Types – Input and Output in C++ (cin, cout) – Operators and Expressions – Control Structures: Decision Making Statements: if, if-else, Nested if, switch – break, continue – Looping Statements: for, while, do-while.										15
II	Functions in C++ – Inline Functions – Function Overloading. Classes and Objects – Defining Classes – Creating Objects – Member Functions – Access Specifiers (public, private, protected) – Constructors: Default and Parameterized Constructors – Destructor – Static Data Members and Static Member Functions – Array of Objects – Friend Functions.										15
III	Operator Overloading – Overloading Unary and Binary Operators – Inheritance – Base and Derived Classes – Types of Inheritance: Single, Multilevel, Multiple, Hierarchical and Hybrid – Virtual Base Class (Introduction) – Abstract Classes (Basics).										15

IV	Pointers – Pointer Declaration – Pointer to Objects – this Pointer – Arrays and Array of Objects – Dynamic Memory Allocation – new and delete Operators – Dynamic Objects – Polymorphism – Function Overriding – Virtual Functions – Runtime Polymorphism.	15
V	File Handling – File Stream Classes – File Modes – Sequential Read and Write Operations – Binary and Text Files – Random Access Files – Exception Handling (try, throw, catch) – String Class – Declaring and Initializing String Objects – String Functions.	15
	Total	75
Course Outcomes		Programme Outcome
CO	Upon completion of the course the students would be able to:	
1	Remember the program structure of C with its syntax and semantics	PO1,PO6
2	Understand the programming principles in C (data types, operators, branching and looping, arrays, functions, structures, pointers and files)	PO2
3	Apply the programming principles learnt in real- time problems	PO4 ,PO7
4	Analyze the various methods of solving a problem and choose the best method	PO6
5	Code, debug and test the programs with appropriate test cases	PO7,PO8
Text Book		
1	E. Balagurusamy, “Object-Oriented Programming with C++”, TMH 2013, 7th Edition.	
Reference Books		
1.	Ashok N Kamthane, “Object-Oriented Programming with ANSI and Turbo C++”, Pearson Education 2003.	
2.	Maria Litvin& Gray Litvin, “C++ for you”, Vikas publication 2002.	
Web Resources		
	https://alison.com/course/introduction-to-c-plus-plus-programming	

Mapping with Program Outcomes:

CO/PSO	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	3	2	1	-	-	1
CO2	2	2	2	1	-	-
CO3	3	1	1	-	1	-
CO4	1	2	1	2	2	1
CO5	3	2	1	2	3	2
Weightage of course contributed to each PSO	12	9	6	5	6	4

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

Title of the Course/ Paper	Subject Name	Category	L	T	P	S	Credits	Inst. Hours	Marks		
									CIA	External	Total
Core Course II	Object Oriented Programming Lab	Core	-	-	Y	-	5	5	40	60	100
Course Objective											
CO1	Describe the procedural and object oriented paradigm with concepts of streams, classes, functions, data and objects										
CO2	Understand dynamic memory management techniques using pointers, constructors, destructors, etc										
CO3	Describe the concept of function overloading, operator overloading, virtual functions and polymorphism										
CO4	Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming										
CO5	Demonstrate the use of various OOPs concepts with the help of programs										
S.No	Details									No. of Hours	
1	Write a C++ program to demonstrate function overloading, Default Arguments and Inlinefunction.										
2	Write a C++ program to demonstrate Class and Objects										
3	Write a C++ program to demonstrate the concept of Passing Objects to Functions										
4	Write a C++ program to demonstrate the Friend Functions.										
5	Write a C++ program to demonstrate the concept of Passing Objects to Functions										
6	Write a C++ program to demonstrate Constructor and Destructor										

7	Write a C++ program to demonstrate Unary Operator Overloading	
8	Write a C++ program to demonstrate Binary Operator Overloading	
9	Write a C++ program to demonstrate: Single Inheritance Multilevel Inheritance Multiple Inheritance Hierarchical Inheritance Hybrid Inheritance	75
10	Write a C++ program to demonstrate Virtual Functions.	
11	Write a C++ program to manipulate a Text File.	
12	Write a C++ program to perform Sequential I/O Operations on a file.	
13	Write a C++ program to find the Biggest Number using Command Line Arguments	
14	Write a C++ program to demonstrate String Manipulation Techniques.	
15	Write a C++ program to demonstrate Exception Handling.	
	Total Hours	75
Course Outcomes		Programme Outcome
CO	Upon completion of the course the students would be able to:	
1	Remember the program structure of C with its syntax and semantics	PO1,PO6
2	Understand the programming principles in C (data types, operators, branching and looping, arrays, functions, structures, pointers and files)	PO2
3	Apply the programming principles learnt in real-time problems	PO4 ,PO7

4	Analyze the various methods of solving a problem and choose the best method	PO6
5	Code, debug and test the programs with appropriate test cases	PO7,PO8
Text Book		
1	E. Balagurusamy, “Object-Oriented Programming with C++”, TMH 2013, 7th Edition.	
Reference Books		
1.	Ashok N Kamthane, “Object-Oriented Programming with ANSI and Turbo C++”, Pearson Education 2003.	
2.	Maria Litvin& Gray Litvin, “C++ for you”, Vikas publication 2002.	
Web Resources		
1.	https://alison.com/course/introduction-to-c-plus-plus-programming	

Mapping with Program Outcomes:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	-	-	-	-
CO2	3	3	-	-	-	-
CO3	3	3	-	-	-	-
CO4	3	3	-	2	-	-
CO5	3	3	1	2	-	2
Weightage of course contributed to each PO	15	14	1	4	0	2

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

Course Code: SEC-1	Office Automation Lab		Credits: 2
Lecture Hours: (L) per week: -	Tutorial Hours: (T) per week	Lab Practice Hours: 3 per week	Total: (L+T+P) per week: 3
Course Category : Skill Enhancement	Year & Semester: I Year I Semester	Admission Year:	
Pre-requisite	Basic skills in Computer operations		
Learning Objectives: (for teachers: what they have to do in the class/lab/field) ● Learn advanced document creation. ● Create professional spreadsheets for business applications. ● Develop attractive presentations. ● Design electronic forms and reports. ● Automate office tasks using Mail Merge and Macros. ● Work with cloud-based office applications. ● Manage PDF documents efficiently.			

Course Outcomes: (for students: To know what they are going to learn) CO1: Create professional documents using advanced word processing features. CO2: Develop spreadsheets with formulas, functions, charts, Pivot Tables, and data analysis techniques. CO3: Design effective multimedia presentations using advanced presentation tools. CO4: Perform office automation tasks such as Mail Merge, PDF management, forms creation, and document security. CO5: Use cloud-based collaboration tools and AI-assisted productivity applications to improve office efficiency and digital workplace skills. Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

S.no	Contents	Required Hours
I	LIST OF PROGRAMS: I. ADVANCED WORD PROCESSING 1. Create professional business letters. 2. Design Resume/CV using templates. 3. Create reports with Table of Contents. 4. Insert Header, Footer and Page Number. 5. Create Tables and Formatting Styles. 6. Use Themes and Templates. 7. Track Changes and Comments. 8. Protect documents with Password. II. SPREADSHEET AUTOMATION 1. Employee Salary Sheet. 2. Student Mark Sheet. 3. Grade Calculation. 4. Budget Preparation. 5. Charts and Graphs. 6. Conditional Formatting. 7. Data Validation. 8. Sorting and Filtering. 9. Pivot Tables. The above program is executed based on the specified given function. • SUM • AVERAGE • COUNT • IF • MAX • MIN • ROUND • CONCAT • TEXT • TODAY • DATE	45 Hours

	III. PRESENTATION DESIGN 1. Educational Presentation. 2. Insert SmartArt. 3. Insert Charts. 4. Slide Master. 5. Animation. 6. Slide Transitions. 7. Presenter View. IV OFFICE PRODUCTIVITY TOOLS 1. Mail Merge (Letters) 2. Digital Signature 3. PDF Creation 4. PDF Password Protection 5. Convert Word ↔ PDF	
--	--	--

Learning Resources:

- **Recommended Texts**

1. Peter Norton, “Introduction to Computers” –Tata McGraw-Hill.

- **Reference Books**

1. Jennifer Ackerman Kettel, Guy Hat-Davis, Curt Simmons, “Microsoft 2003”, Tata McGraw- Hill.

- **Web resources:** Web content from NDL / SWAYAM or open source web resources

Mapping with Program Outcomes:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	2	2	-	1	-	1
CO2	3	3	-	1	-	2
CO3	2	2	1	1	-	2
CO4	2	2	-	2	-	1
CO5	2	2	1	2	1	3
Weightage of course contributed to each PO	11	11	2	7	1	9

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

Course Code: FC	Problem Solving Techniques		Credits: 2
Lecture Hours: (L) per week: 2	Tutorial Hours: (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 2
Course Category : Foundation	Year & Semester: I Year I Semester		Admission Year:
Pre-requisite	Basic of Problem-solving skills		
Learning Objectives: • To understand the importance of algorithms and programs, and to know of the basic problem solving strategies. • To learn efficient strategies and algorithms to solve standard problems, thus laying a firm foundation for designing algorithmic solutions to problems.			
Course Outcomes: (for students: To know what they are going to learn) CO1: Understand the systematic approach to problem solving. CO2: Know the approach and algorithms to solve specific fundamental problems. CO3: Understand the efficient approach to solve specific factoring-related problems. CO4: Understand the efficient array-related techniques to solve specific problems. CO5: Understand the efficient methods to solve specific problems related to text processing. Understand how recursion works.			
Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)			

Units	Contents	Required Hours
I	Introduction: Concept of Algorithms and Programs – Characteristics of an Algorithm – Steps in Problem Solving – Problem Definition and Analysis – Problem-Solving Strategies – Top-Down Design – Algorithm Implementation – Introduction to Recursion.	06
II	Fundamental Algorithms: Swapping Two Variables – Counting – Summation of Numbers – Factorial Computation – Fibonacci Series – Reverse of an Integer – Number Base Conversion – Introduction to Sine Function Computation.	06

III	Number Algorithms: Square Root of a Number – Smallest Divisor – Greatest Common Divisor (GCD) – Prime Number Generation – Prime Factorization – Pseudo-Random Number Generation – Power Computation – Efficient Fibonacci Computation.	06
IV	Array Techniques: Array Traversal – Array Reversal – Finding Maximum Element – Counting Frequency (Histogram) – Removing Duplicate Elements – Partitioning an Array – Finding the kth Smallest Element – Introduction to Longest Monotone Subsequence.	06
V	Text Processing and Recursive Algorithms: Text Formatting – Text Justification – Keyword Searching – Text Editing – Linear Pattern Searching – Recursive Algorithms – Towers of Hanoi – Permutation Generation.	06
	Total hours	30

Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the Course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

1. R. G. Dromey, *How to Solve it by Computer*, Pearson India, 2007.

- **Reference Books**

1. George Polya, Jeremy Kilpatrick, *The Stanford Mathematics Problem Book: With Hints and Solutions*, Dover Publications, 2009 (Kindle Edition 2013).
2. Greg W. Scragg, *Problem Solving with Computers*, Jones & Bartlett 1st edition, 1996.

Mapping with Programme Outcomes

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	3	-	-	-	1
CO2	3	3	-	-	-	1
CO3	3	3	-	-	-	-
CO4	3	3	-	-	-	1
CO5	3	3	1	-	-	1
Weightage of course contributed to each PO	15	15	1	0	0	4

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

Course Code:EC1	STATISTICS		Credits: 3
Lecture Hours: (L) per week: 5	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 5
Course Category :	Year & Semester: I Year I Semester	Admission Year:2026	
Pre-requisite	Basic Knowledge on Statistics and Its applications		
Learning Objectives: (for teachers: what they have to do in the class/lab/field) To understand the mathematical concepts and Business statistics .			
Course Outcomes: (for students: To know what they are going to learn) CO1: Know the uses of statistics in society CO2: Organize, manage and present data CO3: To get an idea on mean, median and mode CO4: Understanding the Concept and importance of measures of dispersion CO5: Able to understand different types of index numbers.			
Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)			
Units	Contents		Required Hours

I	Methods of collection: Sample Survey - Primary data - Secondary data sources - Presentation of Data: Presentation of data by tables - construction of tables (Univariate and Bivariate) – frequency table and contingency table.	
II	Diagrammatic presentation : Diagrammatic representation of data using Line diagram, Bar diagrams: Simple, multiple, subdivided and Percentage bar diagrams -Pie chart – Graphical representation using histograms, frequency polygons curve – Simple problems only	
III	Measures of Central Tendency – Concept and importance of measures of central tendency – Arithmetic Mean, Median, and Mode and their applications in summarizing data – Computation of Mean, Median, and Mode for individual, discrete, and continuous series – Simple problems only.	
IV	Measures of dispersion: Range-Quartile deviation-Co-efficient of range-Coefficient of quartile deviation - mean deviation about mean - standard deviation – simple problems	
V	Index Numbers – Introduction and significance of index numbers – Construction of simple aggregative and simple relative index numbers – Fixed base and chain base index numbers – Wholesale Price Index (WPI) and Cost of Living Index (Consumer Price Index) – Computation and interpretation of index numbers through simple problems only.	

Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

First Year (Semester II)

Course Code-CC3	Python Programming		Credits 5
Lecture Hours: (L) per week – 5	Tutorial Hours: (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 5
Course Category : Core	Year & Semester: II Year III Semester	Admission Year:	
Pre-requisite	Basic Knowledge of Programming concept		
Learning Objectives: (for teachers: what they have to do in the class/lab/field) • Describe the core syntax and semantics of Python programming language. • Discover the need for working with the strings and functions. • Illustrate the process of structuring the data using lists, dictionaries, tuples and sets. • Understand the usage of packages and Dictionaries			
Course Outcomes: (for students: To know what they are going to learn) CO1: Develop and execute simple Python programs CO2: Write simple Python programs using conditionals and looping for solving problems CO3: Decompose a Python program into functions CO4: Represent compound data using Python lists, tuples, dictionaries etc. CO5: Read and write data from/to files in Python programs			
Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)			
Units	Contents		Required Hours
I	Basics of Python Programming: History of Python- Features of Python-Literal-Constants-Variables - Identifiers–Keywords–Built-in Data Types-Output Statements –Input Statements–Comments –Indentation- Operators-Expressions-Type conversions. Python Arrays: Defining and Processing Arrays–Array methods.		15

II	Control Statements: Selection/Conditional Branching statements: if, if-else, nested if and if-else statements. Iterative Statements: while loop, for loop, else suite in loop and nested loops. Jump Statements: break, continue and pass statements.	15
III	Functions: Function Definition – Function Call – Variable Scope and its Life time- Return Statement. Function Arguments: Required Arguments, Keyword Arguments, Default Arguments and Variable Length Arguments Recursion. Python Strings: String operations- Immutable Strings - Built-in String Methods and Functions - String Comparison. Modules: import statement- The Python module – dir() function – Modules and Namespace–Defining our own modules.	15
IV	Lists: Creating a list –Access values in List-Updating values in Lists-Nested Lists-Basic List Operations-List Methods. Tuples: Creating, Accessing, Updating and Deleting Elements in a tuple – Nested tuples –Difference between lists and tuples. Dictionaries and Sets: Dictionary type in Python - Set Data type.	15
V	Python File Handling :Types of files in Python - Opening and Closing files-Reading and Writing files: write() and write lines() methods- append()method– read()and readlines()methods–with keyword– Splitting words –File methods- File Positions-Renaming and deleting files.	15
	Total Hours	75
Extended Professional Component (is a part of internal component only, Not to be	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed	

included in the External Examination question paper) Skills acquired from the course	during the Tutorial hour) Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	
Learning Resources: Recommended Texts - Charles Dierbach, “Introduction to Computer Science using Python - A computational Problem solving Focus”, Wiley India Edition, 2015. - Wesley J. Chun, “Core Python Applications Programming”, 3rd Edition , Pearson Education, 2016 Reference Books - Mark Lutz, “Learning Python Powerful Object Oriented Programming”, O’reilly Media 2018, 5th Edition.		

Mapping with Programme Outcomes:

CO/PSO	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	3	2	2	3	3	3
CO2	3	2	2	3	2	3
CO3	3	2	2	3	2	2
CO4	3	2	2	3	2	3
CO5	3	2	2	3	3	3
Weight age of course contributed to each PSO	15	10	10	15	13	14

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

Course Code: CC4	Python Programming Lab		Credits : 5
Lecture Hours: (L) per week:	Tutorial Hours: (T) per week	Lab Practice Hours: 5 per week	Total: (L+T+P) per week: 5
Course Category :Core	Year & Semester: II Year III Semester	Admission Year:	
Pre-requisite	Basic of programming skill		
Learning Objectives: (for teachers: what they have to do in the class/lab/field) ● Acquire programming skills in core Python. ● Acquire Object-oriented programming skills in Python. ● Develop the skill of designing graphical-user interfaces (GUI) in Python. ● Develop the ability to write database applications in Python. ● Acquire Python programming skills to move into specific branches			
Course Outcomes: (for students: To know what they are going to learn) CO1: To understand the problem solving approaches CO2: To learn the basic programming constructs in Python CO3: To practice various computing strategies for Python-based solutions to real world problems CO4: To use Python data structures - lists, tuples, dictionaries. CO5: To do input/output with files in Python.			
Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)			
Index	List of Exercises:		Required Hours
I	1. Program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon user’s choice. 2. Program to calculate total marks, percentage and grade of a student. Marks obtained in each of the five subjects are to be input by user. Assign grades according to the following criteria: Grade A: Percentage >=80 Grade B: Percentage >=70 and <80 Grade C: Percentage >=60 and <70 Grade D: Percentage >=40 and <60 Grade		75

	E: Percentage < 40 - Program to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user. - Write a Python script that prints prime numbers less than 20. - Program to find factorial of the given number using recursive function. - Write a Python program to count the number of even and odd numbers from array of N numbers. - Write a Python class to reverse a string word by word. - Given a tuple and a list as input, write a program to count the occurrences of all items of the list in the tuple. (Input : tuple = ('a', 'a', 'c', 'b', 'd'), list = ['a', 'b'], Output : 3) - Write a Python program to construct the following pattern, using a nested loop <pre> * ** *** **** ***** ***** ***** *** ** * </pre> - Program using Dictionaries. - Program using Set. - Read a file content and copy only the contents at odd lines into a new file.	
	Total Hours	75

Extended Professional Component (is a part of internal component only, Not to be included in the External Examination Question paper) Skills acquired from the course	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC –CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour) Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill.	
---	--	--

Learning Resources:

● **Recommended Texts**

1. Charles Dierbach, “Introduction to Computer Science using Python - A computational Problem solving Focus”, Wiley India Edition, 2015.
2. Wesley J. Chun, “Core Python Applications Programming”, 3rd Edition , Pearson Education, 2016

● **Reference Books**

1. Mark Lutz, “Learning Python Powerful Object Oriented Programming”, O’reilly Media 2018, 5th Edition.
2. Timothy A. Budd, “Exploring Python”, Tata MCGraw Hill Education Private Limited 2011, 1 st Edition.

Mapping with Programme Outcomes:

CO/PSO	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO1	2	2	2	2	3	2
CO2	2	1	3	2	-	2
CO3	3	3	1	1	1	2
CO4	2	3	3	1	-	1
CO5	3	2	3	1	1	-
Weightage of course contributed to each PSO	12	11	12	7	5	7

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

Course Code: SEC-2	Understanding Internet Technology	Credits: 2
---------------------------	--	-------------------

Lecture Hours: (L) per week: 2	Tutorial Hours: (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 2
Course Category: Skill Enhancement	Year & Semester: I Year II Semester		Admission Year:
Pre-requisite	A willingness to learn and use digital technologies.		
Learning Objectives: (for teachers: what they have to do in the class/lab/field) • To understand the fundamentals of Internet technologies and networking concepts. • To learn Internet communication protocols and web services. • To explore cloud computing and online collaboration tools. • To understand cyber security and safe Internet practices. • To gain knowledge of emerging Internet technologies and applications.			
Course Outcomes: (for students: To know what they are going to learn)			
CO1: Explain the fundamentals of Internet technologies and Internet architecture.			
CO2: Describe Internet communication protocols, web technologies, and online services.			
CO3: Understand cloud computing concepts and online collaboration tools.			
CO4: Identify cybersecurity threats and apply safe Internet practices.			
CO5: Explain emerging Internet technologies and their real-world applications.			
Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)			
Units	Contents		Required Hours
I	Evolution and History of the Internet – Development of the Internet, Internet, Intranet and Extranet, Internet architecture, Client–Server model, Internet Service Providers (ISP), Domain Name System (DNS), IP addressing (IPv4 and IPv6), Uniform Resource Locator (URL), and the role of web browsers.		06
II	Internet Communication Protocols – Overview of the TCP/IP model, HTTP and HTTPS, FTP, SMTP, POP3 and IMAP protocols, email communication and netiquette, search engines and effective search techniques, static and		06

	dynamic websites, introduction to HTML, CSS and JavaScript.	
III	Cloud Computing Fundamentals – Introduction to cloud computing, cloud deployment models (Public, Private and Hybrid), cloud service models (IaaS, PaaS and SaaS), online storage and file sharing, Google Workspace, Microsoft 365, online collaboration platforms and their applications.	06
IV	Cyber Security and Safe Internet Usage – Fundamentals of cyber security, malware (virus, worm, Trojan, spyware and ransomware), password security, multi-factor authentication, firewalls, antivirus software, phishing attacks, social engineering, safe Internet browsing, digital ethics and responsible Internet usage.	06
V	Modern Internet Applications – Introduction to the Internet of Things (IoT), Artificial Intelligence in Internet applications, Big Data concepts, e-commerce, digital payment systems, social media technologies, cloud-based applications and future trends in Internet technology.	06
	Total Hours	30
Learning Resources: Recommended Texts 1. Behrouz A. Forouzan, Data Communications and Networking, McGraw-Hill Education. 2. Douglas E. Comer, <i>The Internet Book</i>, Pearson Education. • Web resources: Mozilla Developer Network (MDN) Web Docs Comprehensive resource for HTML, CSS, JavaScript, HTTP, web APIs, and modern web technologies.		

Mapping with Programme Outcomes:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	-	-	-	-
CO2	3	3	-	-	-	1
CO3	2	2	1	-	1	2
CO4	2	3	2	3	2	1
CO5	2	2	2	1	2	1
Weightage of course contributed to each PO	12	12	5	4	5	5

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

Course Code: SEC-3	AI Automation Lab	Credits: 2
---------------------------	--------------------------	-------------------

Lecture Hours: (L) per week: -	Tutorial Hours: (T) per week	Lab Practice Hours: 3 per week	Total: (L+T+P) per week: 3
Course Category : Skill Enhancement	Year & Semester: I Year I Semester	Admission Year:	
Pre-requisite	Basic skills in Computer operations using AI		
Learning Objectives: (for teachers: what they have to do in the class/lab/field) • To introduce students to Artificial Intelligence concepts. • To use AI tools for creating documents, presentations, images, and simple programs. • To improve productivity using AI-assisted applications. • To understand the ethical use of AI.			

Course Outcomes: (for students: To know what they are going to learn) CO1: Explain the basic concepts and applications of Artificial Intelligence. CO2: Create documents and presentations using AI-assisted tools. CO3: Create spreadsheets using AI-assisted tools. CO4: Develop simple algorithms, flowcharts, and basic programs with AI assistance. CO5: Design creative digital content such as posters, logos, and greeting cards using AI tools.		
Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)		
S.no	Contents	Required Hours

I	LIST OF PROGRAMS: I. INTRODUCTION TO AI AND PROMPT ENGINEERING 1. Introduction to Artificial Intelligence tools. 2. Creating effective prompts using Prompt Engineering. 3. AI-assisted text generation. 4. AI-based grammar correction and content rewriting. II. AI FOR OFFICE AUTOMATION 10. Generate professional resumes using AI. 11. Create reports using AI tools. 12. Generate PowerPoint presentations automatically. 13. AI-assisted spreadsheet creation. 14. AI-generated charts and graphs. III. AI FOR MULTIMEDIA AUTOMATION 1. AI image generation from text prompts. 2. AI logo and poster creation. 3. AI-assisted image editing. 4. AI presentation with narration. IV. WORKFLOW AUTOMATION USING AI 1. Automating repetitive tasks using AI. 2. AI chatbot creation. 3. AI-assisted data analysis.	3 Hours Per Week
----------	--	------------------

Learning Resources:

- **Recommended Texts**

1. Artificial Intelligence: A Modern Approach – Stuart Russell and Peter Norvig, Pearson.

- **Reference Books**

1. Artificial Intelligence – McGraw-Hill.

- **Web resources:** <https://chatgpt.com/> (AI assistant for content generation, coding, summarization, and automation)
- https://www.canva.com/ai/?utm_source=chatgpt.com (AI-powered presentation and image generation)

Mapping with Program Outcomes:

CO/PO	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	1	1	-	-
CO2	2	2	-	2	-	1
CO3	2	3	-	2	-	1
CO4	3	3	-	1	-	2
CO5	2	2	2	2	1	2
Weightage of course contributed to each PO	12	12	3	8	1	6

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

Course Code:EC2	MATHEMATICS		Credits: 3
Lecture Hours: (L) per week: 5	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 5
Course Category :	Year & Semester: I Year II Semester	Admission Year:2026	
Pre-requisite	Basic Knowledge on Set theory and mathematical logic		
Learning Objectives: (for teachers: what they have to do in the class/lab/field) To understand the mathematical concepts like set theory and linear algebra . Course Outcomes: (for students: To know what they are going to learn) CO1: To gain knowledge on set theory CO2: Able to understand Cramer’s rule and its applications CO3: To get an idea on matrices CO4: To gain knowledge on Mathematical logic CO5: Able to understand logical operators and its applications Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) This is done during 2 Tutorial hours)			
Units	Contents		Required Hours

I	Set Theory Sets: Introduction to sets; methods of describing sets, including the roster (tabulation) method and the set-builder method; classification of sets such as singleton sets, null (empty) sets, equal sets, disjoint sets, and universal sets. Set operations: union, intersection, difference, and complement of a set. Venn diagrams and their applications in representing sets and set operations. Solving simple problems involving set operations and simple Venn diagrams.	
II	<i>Matrices:</i> Matrices: Introduction to matrices; classification of matrices including equal matrices, diagonal matrices, scalar matrices, unit (identity) matrices, null (zero) matrices, row matrices and column matrices. Operations on matrices: addition, subtraction and multiplication of matrices. Solution of simple problems involving basic matrix operations.	
III	<i>Matrices:</i> Matrices: Definitions of Inverse of a matrix, determinant of a matrix and Cramer's Rule. Solution of simple problems involving determinants, inverses and Cramer's Rule.	
IV	Mathematical Logic: Introduction to mathematical logic and basic logical concepts. Logical operations including conjunction, disjunction, negation, conditional and Bi-conditional operators, studied through truth tables. Solution of simple problems involving logical operations and truth table construction.	

V	Mathematical Logic: Algebra of propositions -Associative laws, commutative laws, distributive laws, De Morgan's laws- Definition- Tautology and Contradiction-Using only truth tables. simple problems only.	
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	

Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	
---------------------------------	---	--

Learning Resources:

Recommended Texts

- Differential Calculus, S. Chand & Company Ltd.
- Integral Calculus, S. Chand & Company Ltd.

Reference Books

- Allied Mathematics, Margham Publications, Chennai.
- Calculus, S. Viswanathan (Printers & Publishers) Pvt. Ltd., Chennai.

Web resources: Web resources from NDL Library, E-content from open-source libraries