BHARATHIDASAN UNIVERSITY   TIRUCHIRAPPALLI - 620 024.

B.Sc. COMPUTER SCIENCE: CHOICE BASED CREDIT SYSTEM –
LEARNING OUTCOMES-BASED CURRICULUM FRAMEWORK (CBCS - LOCF)

(Applicable to the candidates admitted from the academic year 2022-2023 onwards)

Sem.	Part	Course	Title	Hrs	Credits	Exam Hours	Marks Int.	Marks Ext.	Total
I	I	Language Course – I (Tamil $/Other Languages + #)		6	3	3	25	75	100
	II	English Course - I		6	3	3	25	75	100
	III	Core Course – I (CC)	Programming in C and Data Structures	5	5	3	25	75	100
		Core Practical – I (CP)	Programming in C Lab	4	4	3	40	60	100
		First Allied Course – I (AC)		4	4	3	25	75	100
		First Allied Course – II (AC)		3	-	-	-	-	-
	IV	Value Education		2	2	3	25	75	100
		TOTAL		**30**	**21**	**-**	**-**		**600**
II	I	Language Course - II (Tamil $/Other Languages + #)		6	3	3	25	75	100
	II	English Course - II		6	3	3	25	75	100
	III	Core Course – II (CC)	Programming in Java	5	5	3	25	75	100
		Core Practical – II (CP)	Programming in Java Lab	4	4	3	40	60	100
		First Allied Course – II (AC)		3	2	3	25	75	100
		First Allied Course – III (AC)		4	4	3	25	75	100
	IV	Environmental Studies		2	2	3	25	75	100
		TOTAL		**30**	**23**	**-**	**-**	**-**	**700**

$ For those who studied Tamil up to 10th +2 (Regular Stream)

+ Syllabus for other Languages should be on par with Tamil at degree level

Those who studied Tamil up to 10th +2 but opt for other languages in degree level under Part I should study special Tamil in Part IV

* Extension Activities shall be outside instruction hours.

List of Allied Courses

Allied Course I   Allied Course II

Mathematics   **Applied Physics**

1

SUMMARY OF CURRICULUM STRUCTURE OF UG PROGRAMMES

Sl. No.	Part	Types of the Courses	No. of Courses	No. of Credits	Marks
1.	I	Language Courses	4	12	400
2.	II	English Courses	4	12	400
3.	III	Core Courses	9	45	900
4.		Core Practical	6	24	600
5.		Allied Courses I & II	4	16	400
6.		Allied Practical	2	4	200
7.		Major Based Elective Courses	2	8	200
8.		Project	1	3	100
9.	IV	Non-Major Elective Courses	2	4	200
10.		Skill Based Elective Courses	2	4	200
11.		Soft Skills Development	1	2	100
12.		Value Education	1	2	100
13.		Environmental Studies	1	2	100
14.	V	Gender Studies	1	1	100
15.		Extension Activities	1	1	---
16.		**Total**	**41**	**140**	**4000**

PROGRAMME OUTCOMES:

- Graduates will be able to comprehend the basic concepts learnt and apply in real life situations with analytical skills.
- Graduates with acquired skills and enhanced knowledge will be employable / become entrepreneurs or will pursue higher Education.
- Graduates with acquired knowledge of modern software tools will be able to contribute effectively as software engineers.
- Graduates will be able to comprehend the related concepts to Computer Science with Allied papers
- Graduates will be imbibed with ethical values and social concerns to ensure peaceful society.

PROGRAMME SPECIFIC OUTCOMES:

- Acquired the required knowledge in the Hardware and Software aspects of Computer Science domain and the art of programming.
- Understood the development methodologies of software systems and the ability to analyze, design and develop computer applications for real life problems.
- Gained knowledge and skills to collaborate and communicate with peers in IT / ITES industries
- The ability to understand, adjust and adapt with the dynamic technical environment for the growth of IT industry.
- The capacity to transfer the skills gained, to provide innovative and novel solutions by maintaining ethical norms for the betterment of humane society.

*****

First Year	**CORE COURSE I**	**Semester I**

**PROGRAMMING IN C AND
DATA STRUCTURES**

Code:	**(Theory)**	**Credit: 5**

COURSE OBJECTIVES:

• To know about the basics of C Programming, Control and Looping Structures and programming with it.
• To understand Arrays, Pointers and String Processing in C language
• To know about the basic concepts in Data Structures.

UNIT - I:

Basic of C: History of C and its importance – Structure of a C program – Data Types – Constants and Variables – Operators and Expressions – Order of Precedence, Evaluating of Arithmetic Expressions – Type Conversion- Decision Statements: if, if-else, and nested if statements.

UNIT - II:

Loops Structures: For Loop, While, Do-while loop – Arrays: - One Dimensional Array, Two-dimensional Arrays, Character Arrays and Strings – Functions: Function with arrays- Function with decision and looping statements - Recursion.

UNIT - III:

Pointers: Introduction – Pointer Expressions – Chain of Pointers –Pointers and Arrays – Array of Pointers – Pointers as function arguments – Functions returning Pointers – Pointers to Functions – Function pointer – Structures - declaration, initialization, Array of Structures – Pointer to structures, Structures and functions – Typed of Enumerated data types, Unions.

UNIT - IV:

Strings Processing, Standard string library functions – Files: introduction and files functions – Writing and reading in Text mode – Simple application: Display the contents of a file. Write data to a file. Append data to an existing file – File IO – Reading and writing structures.

UNIT - V:

Stack: LIFO concept, Stack operations, Array implementation of stack – Queue: FIFO concept, Queue operations, Array implementation of queue – Singly Linked List: concepts, operations – Doubly Linked List: concepts, operations – Trees: General trees, Binary trees.

UNIT VI: CURRENT CONTOURS (for Continuous Internal Assessment Only):

Contemporary Developments Related to the Course during the Semester Concerned.

REFERENCE BOOKS:

1. E. Balagurusamy, "Programming in ANSI C", Tata McGraw Hill, New Delhi, Seventh Edition, 2016.
2. E.Horowitz, S.Sahni and Susan Anderson Freed, "Fundamental Data Structures in C", 2ed, Orient BlackSwan Publisher, 2009.
3. Byron S. Gottfried, "Programming with C", Schaum's Outline Series, Tata-McGraw Hill Edition, New Delhi, 1991.
4. E. Karthikeyan, "A Textbook on C Fundamentals, Data Structures and Problem Solving",Prentice-Hall of India Private Limited, New Delhi, 2008.
5. YashavantKanetkar, "Let us C", BPB Publications, Tenth Edition, New Delhi, 2010.
6. Szuhay, Jeff, and Szuhay, Jeff, "Learn C Programming: A Beginner's Guide to Learning C Programming the Easy and Disciplined Way", Packt Publishing, 2020.
7. Jena, Sisir Kumar, and Jena, Sisir Kumar, "C Programming: Learn to Code", CRC Press, 2021.
8. https://www.tutorialspoint.com/cprogramming/index.htm
9. https://www.w3schools.in/data-structures/intro

COURSE OUTCOMES:

Upon successful completion of this course the students would be able to:

• Summarize the basic knowledge to develop C programs
• Manipulate Looping, arrays and functions
• Apply and write programs for solving real world problems
• Create open, read, manipulate, write and close files.
• Understand the basic concepts in data structures.

*****

First Year   **CORE PRACTICAL I**   **Semester I**
                          PROGRAMMING IN C LAB
Code: **(Practical)**   **Credit: 4**

COURSE OBJECTIVES:

- To understand the programming fundamentals of C language.
- To impart writing skill of C programming and data structures for a list of problems.
- To impart hands-on training for writing a C program using computers.

1. Write a Program
   (i) To convert temperature from degree Centigrade to Fahrenheit,
   (ii) Find whether given number is Even or Odd,
   (iii) Find the greatest of Three numbers.
2. Write a Program to display Monday to Sunday using switch statement
3. Write a Program to display first Ten Natural Numbers and their sum.
4. Write a Program to perform Multiplication of Two Matrices.
5. Write a Program
   (i) To find the maximum number in an Array using pointer.
   (ii) To reverse a number using pointer.
   (iii) To add two numbers using pointer.
6. Write a Program to solve Quadratic Equation using functions.
7. Write a Program to find factorial of a number using Recursion.
8. Write a Program to demonstrate Call by Value and Call by Reference.
9. Write a Program to create a file containing Student Details.
10. Write a program to implement a stack using singly linked list, Implement Queue using Linked List.

COURSE OUTCOMES:

Upon successful completion of this course the students would be able to:

- Relate the use of language constructs to solve simple programs
- Develop programs for various concepts in C language
- Understand and trace the execution of the list of programs
- Understand the usage of file handling in C programming
- Solve data problems related to data structures.

*****

First Year   **CORE COURSE II**   **Semester II**

PROGRAMMING IN JAVA

Code: **(Theory)**   **Credit: 5**

COURSE OBJECTIVES:

- To acquire the programming skills with java.
- To implement the object-oriented concepts with java language
- To learn the art of GUI programming with Applet.

UNIT - I:

Foundation, Essentials, Control Statement and Classes & Objects, Stage of Java – origin of Java – challenges - features - Object-Oriented Programming; Java Essentials: Elements - API - variables - primitive data types – String Class - operators –combined assignment operators - conversion –scope – comments - keyboard input; Control Statements: if ,if-else, nested if & if-else-if statements – logical operators – comparison – conditional operator – switch – increment and decrement – while, do-while & for loops – nested loops – break and continue; Classes and Objects: classes and objects -modifiers - passing arguments– constructors - package & import - static class members –method overloading– constructor overloading –returning objects – this variable – recursion – nested & inner classes – abstract classes & methods.

UNIT - II:

Arrays, String Handling, Inheritance, Interface and Packages, Introduction –processing array – passing arrays – returning arrays – String arrays – two Dimensional Arrays - Arrays with Three or More Dimensions; String Handling : String class – concatenation – comparison – substring – methods – other methods–String Buffer, String Builder & String Tokenizer classes; Inheritance: basics –inheriting and overriding superclass methods – calling superclass constructor – polymorphism – inherit from different classes – abstract classes – final Class; Interfaces: Basics – multiple Interfaces – multiple inheritance using interface – multilevel interface – Packages – Create and access packages in Net Beans IDE – static Import and package class – access specifiers.

UNIT - III:

Exception Handling, I/O and File Handling and Multithreading, Introduction - try and catch block - multiple catch block - nested try - finally Block – throw Statement – exception propagation – throw Clause - custom exception – built-in exception; Multithreading: Introduction – threads – thread creation – life cycle – joining a thread – scheduler &priority – synchronization – inter-thread communication – thread control – thread Pool – thread group – daemon thread; Files and I\O Streams: file Class – streams – byte streams – filtered byte streams – Random Access File class – character streams.

UNIT - IV:

Applet and GUI Part I, Fundamentals – applet class – life cycle – steps for applet program – passing values through parameters – graphics – event handling; GUI I:GUI – creating windows – dialog boxes – layout managers – AWT component classes – Swing component classes – applications of AWT controls.

UNIT - V:

GUI Part II and Java Database Connectivity, Event handling – AWT components – AWT graphics classes – Swing controls – application using Swing and AWT; Java Database Connectivity: types of drivers – JDBC architecture – JDBC classes &interfaces – steps in JDBC applications – creating a new Database and table with JDBC.

UNIT VI: CURRENT CONTOURS (for Continuous Internal Assessment Only):

Contemporary Developments Related to the Course during the Semester Concerned.

REFERENCES:

1. S.Sagayaraj, R.Denis, P.Karthik & D.Gajalakshmi, "Constructive Java Programming", Universities Press, 2021.
2. E. Balagurusamy, "Programming with JAVA", Tata McGraw Hill, New Delhi, 2019.
3. C. Muthu, "Programming with JAVA", Vijay Nicole Imprints Private Limited, Chennai, Second Edition, 2011.
4. Bruce Eckel, Chuck Allison, "Thinking in Java", Prentice Hall Publications, 2006
5. Malina Pronto, "Java: How To Learn Java Programming: How To Improve Your Java Coding In 2020/2021: 5 Programming Languages To Learn For Beginners In Tech", Independently Published, 2020.
6. Nick Samoylov, "Learn Java 12 Programming: A Step-by-step Guide to Learning Essential Concepts in Java", Packt Publishing, 2019.
7. https://www.javatpoint.com/java-tutorial

COURSE OUTCOMES:

Upon successful completion of this course the students would be able to:

- Understand the concept of OOP as well as the purpose and usage principles of inheritance, polymorphism, encapsulation and method overloading.
- Identify members of a class and to implement them
- Create Java application programs using sound OOP practices (e.g., interfaces and APIs) and proper program structuring (e.g., by using access control identifies, and create user define package for specific task,(reusability concepts) error exception handling)
- Develop programs using the Java standard class library.
- Develop software using Java programming language, (using applet, AWT controls, and JDBC).

*****

PROGRAMMING IN JAVA LAB

COURSE OBJECTIVES:

- To understand the basics of JAVA programs and their execution.
- To learn concepts like inheritance, packages and interfaces.
- To understand the life cycle of the applets, database connectivity and their functionality.

1. Write a program to sort the given numbers using arrays.
2. Write a program to implement the FIND and REPLACE operations in the given text.
3. Write a program to implement a calculator to perform basic arithmetic Operations, doing with constructers
4. Write a program to find the student's percentage and grade using command line arguments.
5. Write a program to draw circle or triangle or square using polymorphism and inheritance.
6. Implement multiple inheritance concepts in java using interface, you can choose your own example of a company or education institution or a general concept which requires the use of interface to solve a particular problem.
7. Write a program to create threads and perform operations like start, stop, suspend, resume
8. Write a program to develop an applet to play multiple audio clips using multithreading.
9. Write a program to retrieve employee data from a file
10. Write a program to retrieve student data from a Database

Course Outcomes:

Upon successful completion of this course the students would be able to:

1. Develop java programs to understand the OOP concepts.
2. Write java programs for classes and objects
3. Develop simple programs with multiple threads
4. Write java programs using Applets
5. Develop java programs to connect databases and files.

*****