BHARATHIDASAN UNIVERSITY TIRUCHIRAPPALLI-620 024.

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

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

Revised as on 05.01.2023

Sem.	Part	Course	Title	Ins. Hrs	Credit	Exam Hours	Marks		Total
							Int.	Ext.	
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)	Mathematics - I	4	4	3	25	75	100
		First Allied Course (AP)	Mathematics - II	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)	Python Programming	5	5	3	25	75	100
		Core Practical – II (CP)	Python Programming Lab	4	4	3	40	60	100
		First Allied Course (AP)	Mathematics - II	3	2	3	40	60	100
		First Allied Course – II (AC)	Mathematics - III	4	4	3	25	75	100
		Add on Course – I ##	Professional English- I	*6	4	3	25	75	100
	IV	Environmental Studies		2	2	3	25	75	100
	TOTAL			**30+6***	**27**	**-**	**-**	**-**	**800**

$ For those who studied Tamil upto 10th +2 (Regular Stream

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

Those who studied Tamil upto 10th +2 but opt for other languages in degreelevel under Part I should study special Tamil in Part IV

The Professional English – Four Streams Course is offered in the 2nd and 3rd Semester (only for 2022-2023 Batch) in all UG Courses. It will be taught apart from the Existing hours of teaching/additional hours of teaching (1 hour /day) as a 4 credit paper as an add on course on par with Major Paper and completion of the paper is must to continue his/her studies further. (As per G.O. No. 76, Higher Education (K2) Department dated: 18.07.2020).

*** The Extra 6 hrs./Cycle as per the G.O. 76/2020 will be utilized for the Add on Professional English Course.**

@ NCC Course is one of the Choices in Non-Major Elective Course. Only the NCC cadets are eligible to choose this course. However, NCC Course is not a Compulsory Course for the NCC Cadets.

** Extension Activities shall be outside instruction hours.

List of Allied Courses

First Allied Course I	Second Allied Course II
Mathematics	**Statistics**

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.		**Add –on Course (Professional English I & II)**	2	8	200
9.		Project	1	3	100
10.	IV	Non-Major Elective Courses	2	4	200
11.		Skill Based Elective Courses	2	4	200
12.		Soft Skills Development	1	2	100
13.		Value Education	1	2	100
14.		Environmental Studies	1	2	100
15.	V	Gender Studies	1	1	100
16.		Extension Activities	1	1	---
	Total		**43**	**148**	**4200**

PROGRAMME OBJECTIVES:

- 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 data scientist.
- Graduates will be able to comprehend the related concepts to Data Science with Allied papers
- Graduates will be imbibed with ethical values and social concerns to ensure peaceful society.

PROGRAMME OUTCOMES:

After successful completion of B.Sc. Data Science programme the students will be able to

- Apply the computing knowledge to analyze real world applications.
- Solve the complex problems in the field of data science with an understanding of different types and formats of data
- Understand the concepts and ability to apply appropriate models.
- Develop programs and models for the various domain specific problems.
- The ability to exposure, identify and 3nalyse big data, adjust and adapt with the dynamic technical environment for the growth of IT industry.

*****

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 PRACTICL 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 ofproblems.
- 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**

PYTHON PROGRAMMING

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

COURSE OBJECTIVES:

To provide a platform to learn the fundamentals of Object Oriented Programming

- To understand the syntax of the Python and apply the concepts to write programs
- To implement file concepts and data analysis in Python

UNIT - I:

Object Oriented Programming: Procedural and Object-Oriented Programming – Classes – Working with instances – techniques for designing classes. – Inheritance: introduction to inheritance – Polymorphism

UNIT - II:

Python: Installing python- The python Interpreter – Interactive mode – Writing and running programs in script mode- IDLE programming environment – Input, processing and output – Displaying output with print function –Strings and String literals- Comments – variables – Reading input from the Keyboard - Operators- more about output – Decision structures and Boolean logic – Repetition Structures

UNIT - III:

Lists, Tuples, Strings, Dictionaries and Set: Sequences – Introduction to Lists – List slicing – 'in' operator – list methods and built-in-functions – copying lists – processing lists – Two Dimensional Lists – Tuples. Strings**:** Basic String Operations – String Slicing Testing, Searching and manipulating strings – Dictionaries and Set: Dictionaries – Set – Serializing Objects .

UNIT - IV:

Functions, Modules and File Handling: Functions: introduction to functions – Defining and calling functions – designing a program to use functions – Local variables – passing arguments to functions – Global variable and Global Constants- –Value returning functions: generation – user defined value returning functions – Modules: math module- Storing functions in modules - Files: Introduction to File Input and Output – Using Loops to process files – processing records - Exceptions – Python Standard Library - Regular Expression..

UNIT - V:

Data Analysis using Python: Load data into a Data Frame - Fundamentals of Data Manipulation with Python.

Unit - VI Current Contours (for Continuous Internal Assessment Only):

Contemporary Developments Related to the Course during the Semester Concerned.

REFERENCES:

1. Allen B. Downey, ***"Think Python: How to Think Like a Computer Scientist"***, 2nd edition, Updated for Python 3, Shroff/O'Reilly Publishers, 2016
2. Satyanarayana, Radhika Mani, Jagadesh***,"Python Programming",*** Universities Press (India) Pvt. Ltd 2018
3. Wesley Chun **"Core python Programming"** Pearson Education, 2006,
4. Al Sweigart, **"Invent your own computer games with python",** 2nd edition, 2008
5. Bill Lubanovic, **"Introducing Python"**, O"Reilly, First Edition-Second Release, 2014.
6. Tony Gaddis, **"Starting out with python",** 2nd edition, Addison Wesley, Pearson
7. Michael Dawson, "**Python programming for the absolutebeginner"**, Premier press, 2003.

COURSE OUTCOMES:

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

- Interpret and manipulate the OOPs Concepts
- Install python and write programs to solve simple problems
- Explain basic data structures in Python
- Store and manipulate data using file system
- Implement Python packages and libraries

*****

First Year **CORE PRACTICAL II** **Semester II**

PYTHON PROGRAMMING LAB

Code: **(Practical)** **Credit: 4**

COURSE OBJECTIVES:

- To write, test, and debug simple Python programs.
- To implement Python programs with conditionals and loops.
- To represent compound data using Python lists, tuples, and dictionaries.

1. Write Python programs for the following: (using Basics of Python)
   a. Purposefully raise Indentation Error and correct it.
   b. Compute distance between two points taking input from the user (use Euclidean distance formula).
   c. To takes numbers as command line arguments and print its sum
2. Write Python programs for implementing the following: (using Control Flow)
   a. Finding the factorial of a number.
   b. Print the prime numbers below 100
3. Write Python programs for implementing the following: (using Strings)
   a. Count the numbers of characters in the string and store them in a dictionary data structure
   b. Using split and joins methods in the string and trace a birthday with a dictionary data structure.
4. Write Python programs for the following: (using List)
   a. Finding mean, median, mode for the given set of numbers in a list.
   b. Function dups to find all duplicates in the list.
5. Write Python programs to do the following (using Methods)
   a. Create a Python class named Circle constructed by a radius and two methods which will compute the area and the perimeter of a circle.
   b. Create a Python class named Rectangle constructed by a length and width and a method which will compute the area of a rectangle.
6. Write Python programs to implement
   a. Constructors to assign the PI_VALUE.
   b. Polymorphism to print area of square/rectangle depending upon the number of parameters passed.
7. Write Python programs to implement
   a. Inheritance
   b. Method overloading and overriding
8. Write a python program (using files)
   a. to open and write “Hello World” into a file.
   b. to write the content “Hi Python Programming” for the existing file.
   c. To import values from a CSV file to create Pandas Data Frame
9. Write a Python program to create an Email slicer.
10. Write a Python program to generate password.

COURSE OUTCOMES:

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

1. Write simple programs using control structures, functions and strings
2. Develop programs using tuples, lists, sets and dictionary
3. Write simple programs using Constructors, Method overloading and inheritance
4. Develop programs using files and regular expressions
5. Write simple programs using packages and exception handling

*****