M.Sc. COMPUTER SCIENCE

LOCF SYLLABUS 2023

Department of Information Technology
School of Computing Sciences
St. Joseph's College (Autonomous)
Tiruchirappalli - 620 002, Tamil Nadu, India

Vision

Forming globally competent, committed, compassionate and holistic persons, to be men and women for others, promoting a just society.

Mission

- Fostering learning environment to students of diverse background, developing their inherent skills and competencies through reflection, creation of knowledge and service.
- Nurturing comprehensive learning and best practices through innovative and value-driven pedagogy.
- Contributing significantly to Higher Education through Teaching, Learning, Research and Extension.

Programme Educational Objectives (PEOs)

- Graduates will be able to accomplish professional standards in the global environment.
- Graduates will be able to uphold integrity and human values.
- Graduates will be able to appreciate and promote pluralism and multiculturalism in working environment.

Programme Outcomes (POs)

1. Graduates will be able to apply assimilated knowledge to evolve tangible solution to emerging problems.
2. Graduates will be able to analyze and interpret data to create and design new knowledge.
3. Graduates will be able to engage in innovative and socially relevant research and effectively communicate the findings.
4. Graduates will become ethically committed professional and entrepreneurs upholding human values.
5. Graduates imbibed with ethical values and social concern will be able to understand and appreciate cultural diversity, social harmony and ensure sustainable environment.

Programme Specific Objectives (PSOs)

1. Acquire fundamental knowledge in problem solving, general computing and comprehensive knowledge in Computer Science.
2. Competence to identify, analyze, design, optimize and implement system solutions using contemporary computing techniques which propels towards employability.
3. Gain fundamental knowledge in computational methods and tools for solving real- time problems and implanting the quest for continual learning of novel and in- demand skills.
4. Demonstrate the ability to act as a leader, or as a part of a team to create multi- functional Software Solutions.
5. Ability to showcase discrete practical experiences by implementing various strategies that utilizes a variety of software techniques that are ethical and would be beneficial to the society

CONTINUOUS INTERNAL ASSESSMENT
Categorizing Outcome Assessment Levels Using Bloom's Taxonomy

Level	Cognitive Domain	Description
K1	Remember	It is the ability to remember the previously learned concepts or ideas.
K2	Understand	The learner explains concepts or ideas.
K3	Apply	The learner uses existing knowledge in new contexts.
K4	Analyse	The learner is expected to draw relations among ideas and to compare and contrast.
K5	Evaluate	The learner makes judgements based on sound analysis.
K6	Create	The learner creates something unique or original.

Question Paper Blueprint for Mid and End Semester Tests

<table>
<tr><td colspan="3">Duration: 2 Hours</td><td colspan="6">Maximum Marks: 60</td></tr>
<tr><td colspan="2" rowspan="2">Section</td><td colspan="6">K level*</td><td rowspan="2">Marks</td></tr>
<tr><td>K1</td><td>K2</td><td>K3</td><td>K4</td><td>K5</td><td>K6</td></tr>
<tr><td colspan="2">A (no choice)</td><td>7</td><td></td><td></td><td></td><td></td><td></td><td>7 × 1 = 7</td></tr>
<tr><td colspan="2">B (no choice)</td><td></td><td>5</td><td></td><td></td><td></td><td></td><td>5 × 3 = 15</td></tr>
<tr><td colspan="2">C (either… or type)</td><td></td><td></td><td>3</td><td></td><td></td><td></td><td>3 × 6 = 18</td></tr>
<tr><td rowspan="4">D (2 out of 3)</td><td>Courses with K4 as the highest cognitive level</td><td></td><td></td><td></td><td>2</td><td></td><td></td><td rowspan="4">2 × 10 = 20</td></tr>
<tr><td>Courses with K5 as the highest cognitive level wherein one question each on K4 and K5 is compulsory.
(Note:K4 has two questions whereas, K5 has no choice.)</td><td></td><td></td><td></td><td>1</td><td>1</td><td></td></tr>
<tr><td rowspan="2">Courses with K6 as the highest cognitive level wherein one question each on K5 and K6 is compulsory.
(Note: Mid Sem: K4 has two questions whereas, K5 has no choice; End sem: K5 has two questions whereas, K6 has no choice)</td><td></td><td></td><td></td><td colspan="2">Mid Sem</td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="2">End Sem</td></tr>
<tr><td colspan="6">Total</td><td>60</td></tr>
</table>

** K4 and K5 levels will be assessed in the Mid semester test whereas K5 and K6 levels will be assessed in the End semester test.*

Question Paper Blueprint for Mid and End Semester Tests *(For quantitative courses only)*

Duration: 2 Hours							**Maximum Marks: 60**
Section	**K level**						**Marks**
	K1	**K2**	**K3**	**K4**	**K5**	**K6**	
A *(no choice)*	5	4					9 × 1 = 9
B *(either… or type)*			2	1			3 × 5 = 15
C *(2 out of 3)*					1	1*	2 × 18 = 36
						Total	**60**

NOTE: *K4 and K5 will be assessed in the Mid semester test whereas K5 and K6 will be assessed in the End semester test.*
** K6 compulsory*

SEMESTER EXAMINATION
Question Paper Blueprint for Semester Examination

Duration: 3 Hours								**Maximum Marks: 100**
Section		**K level**						**Marks**
		K1	**K2**	**K3**	**K4**	**K5**	**K6**	
A *(no choice, two questions from each unit)*		10						10 × 1 = 10
B *(no choice, two questions from each unit)*			10					10 × 3 = 30
C *(either… or type, one question from each unit)*				5				5 × 6 = 30
D *(3 out of 5, one question from each unit)*	*Courses with K4 as the highest cognitive level*				3			3 × 10 = 30
	Courses with K5 as the highest cognitive level wherein two K4 questions and one K5 question are compulsory. (Note: Three questions on K4 and two questions on K5)				2	1		
	Courses with K6 as the highest cognitive level wherein one question each on K4, K5, and K6 is compulsory. (Note: Two questions each on K4 and K5 and one question on K6)				1	1	1	
							Total	**100**

Question Paper Blueprint for Semester Examination *(For quantitative courses only)*

Section	Marks	K level
A	10 × 1 = 10	K1
B	5 × 6 = 30 *(either…or)*	K2 *(Q. No. 11 & 12)* K3 *(Q. No. 13, 14 & 15)*
C	4 × 15 = 60 *(4 out of 5)*	K4 *(Q. No. 16 & 17)* K5 *(Q. No. 18 & 19)* K6 *(Q. No. 20 compulsory)*
Total Marks: 100		

Evaluation Pattern for Part IV One/Two Credit Courses

Title of the Course	CIA	Semester Examination	Total Marks
Internship	100		**100**
UG Skill Enhancement Course (Non Major Elective) Foundation Course **PG** Ability Enhancement Course	20 + 10 + 20 = 50	50 *(External member from the Department)*	**100**
Value Education	50	50 *(CoE)*	**100**

PROGRAMME PATTERN			
Semester 1			
Course Code	**Title of the Course**	**Hours**	**Credits**
23PCS1CC01	**Core Course - 1:** Analysis and Design of Algorithms	**6**	**5**
23PCS1CC02	**Core Course - 2:** Object Oriented Analysis and Design and C++	**6**	**5**
23PCS1CP01	**Core Practical - 1:** Algorithm and OOPS	**6**	**4**
23PCS1ES01	**Elective - 1:** Advanced Software Engineering	**5**	**3**
23PCS1ES02	**Elective - 2:** Python Programming	**5**	**3**
23PCS1AE01	**Ability Enhancement Course:** Big Data Analytics	**2**	**1**
	Total	**30**	**21**

Semester	Course Code	Title of the Course	Hours/ Week	Credits
1	23PCS1CC01	**Core Course - 1:** Analysis and Design of Algorithms	6	5

Course Objectives
Enable the students to learn the Elementary Data Structures Algorithms
Presents an Introduction to the Algorithms, their analysis and design
Discuss various methods like Basic Traversal and Search Techniques, Divide and Conquer method, Dynamic programming, Backtracking
Understood the Various Design And Analysis of the algorithms

UNIT I: Introduction **(18 Hours)**

Introduction: - Algorithm Definition and Specification – Space complexity - Time Complexity Asymptotic Notations - Elementary Data Structure: Stacks and Queues – Binary Tree - Binary Search Tree - Heap – Heap sort- Graph.

UNIT II: Traversal and Search Techniques **(18 Hours)**

Basic Traversal And Search Techniques: Techniques for Binary Trees - Techniques for Graphs - Divide and Conquer: - General Method – Binary Search – Merge Sort – Quick Sort.

UNIT III: Greedy Method **(18 Hours)**

The Greedy Method:-General Method – Knapsack Problem – Minimum Cost Spanning Tree – Single Source Shortest Path.

UNIT IV: Dynamic Programming **(18 Hours)**

Dynamic Programming - General Method – Multistage Graphs–All Pair Shortest Path – Optimal Binary Search Trees – 0/1 Knapsacks – Traveling Salesman Problem – Flow Shop Scheduling.

UNIT V: Backtracking **(18 Hours)**

Backtracking:-General Method – 8-QueensProblem – Sum Of Subsets – Graph Coloring – Hamiltonian Cycles – Branch And Bound: - The Method – Traveling Sales person.

Teaching Methodology	Videos, PPT, Demonstration and creation of models

Books for Study

1. Aho, A. V., Hopcroft, J. E., & Ullman, J. D. (2009). *Data Structures and Algorithms*. Addison -Wesley**.**

2. Horowitz, E., & Sahni, S. (1978). *Fundamentals of Computer Algorithms*. Universities Press.

Books for Reference

1. Goodrich. (2003). *Data structures & algorithms in Java* (3rd ed.). Wiley.
2. Skiena. (2008). *The algorithm design manual* (2nd ed.). Springer.
3. Levith, A. (2003). *Introduction to the design and analysis of algorithm*. Pearson Education Asia.

Course Outcomes		
CO No.	**CO-Statements**	**Cognitive Levels**
	On completion of this course, students will,	**(K - Level)**
CO1	demonstrate specific search and sort algorithms using divide and conquer technique	**K1**
CO2	get knowledge about algorithms and determine that time complexity	**K2**
CO3	gain a good understanding of Greedy Method and Its algorithm	**K3**
CO4	be able to describe graphs using dynamic programming techniques	**K4**
CO5	demonstrate the concept of backtracking branch and bound technique	**K5**
CO6	compare different sorting and searching techniques	**K6**

Relationship Matrix

Semester	Course code	Title of the Course	Hours	Credits
1	23PCS1CC01	**Core Course - 1:** Analysis and Design of Algorithms	6	5

Course Outcomes	Programme Outcomes (POs)					Programme Specific Outcomes (PSOs)					Mean Score of COs
	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3	PSO4	PSO5	
CO1	2	3	3	2	1	3	2	2	1	3	2.1
CO2	3	2	2	3	2	1	3	3	2	1	2.2
CO3	3	2	3	3	3	3	2	1	3	1	2.4
CO4	1	2	1	1	3	2	3	3	1	3	2.0
CO5	3	1	2	1	3	2	3	3	3	2	2.3
CO6	2	3	3	2	2	1	3	3	1	2	2.2
									Mean overall Score		2.2 (High)

Semester	Course Code	Title of the Course	Hours/ Week	Credits
1	23PCS1CC02	**Core Course - 2:** Object Oriented Analysis and Design and C++	6	5

Course Objectives
Present The Object model, classes and objects, object orientation, machine view and model management view
Enables the students to learn the basic functions, principles and concepts of object-oriented analysis and design
Enable the students to understand C++ language with respect OOAD

UNIT I: Object Model **(18 Hours)**

The Object Model: The Evolution of the Object Model – Elements of the Object Model – Applying the Object Model. Classes and Objects: The Nature of an Object – Relationship among Objects**.**

UNIT II: Classes and Objects **(18 Hours)**

Classes and Object: Nature of Class – Relationship Among Classes – The Interplay of Classes and Objects. Classification: The importance of Proper Classification – Identifying classes and objects – Key Abstractions and Mechanism.

UNIT III: C++ Introduction **(18 Hours)**

Introduction to C++ - Input and output statements C++ - Declarations - Control Structures – Functions in C++.

UNIT IV: Inheritance and Overloading **(18 Hours)**

Classes and Objects – Constructors and Destructors – Operators Overloading – Type Conversion Inheritance – Pointers and Arrays.

UNIT V: Polymorphism and Files **(18 Hours)**

Memory Management Operators - Polymorphism – Virtual functions – Files – Exception Handling – String Handling - Templates.

Teaching Methodology	Videos, PPT, Demonstration and creation of models

Books for Study

1. Booch, G. (1998). *Object oriented analysis and design with applications* (2nd ed.). Pearson Education.

2. Kamthane, A. N. (2003). *Object-oriented programming with ANSI & Turbo C++*. First Indian Print, Pearson Education.

Books for Reference

1. Balagurusamy. (2003). *Object oriented programming with C++* (2nd ed.). TMH.

Course Outcomes		
CO No.	**CO-Statements**	**Cognitive Levels**
	On completion of this course, students will	
CO1	understand the concept of Object-Oriented development and modeling techniques.	**K1**
CO2	gain knowledge about the various steps performed during object design.	**K2**
CO3	abstract object-based views for generic software systems.	**K3**
CO4	link OOAD with C++ language.	**K4**
CO5	apply the basic concept of OOPs and familiarize to write C++ program.	**K5**
CO6	show the behaviour of exception handling and the streams	**K6**

Relationship Matrix											
Semester	**Course code**	**Title of the Course**								**Hours**	**Credits**
1	**23PCS1CC02**	**Core Course - 2:** Object Oriented Analysis and Design and C++								**6**	**5**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Score of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	**2**	**3**	**3**	**2**	**1**	**3**	**2**	**2**	**1**	**3**	**2.1**
CO2	**3**	**2**	**2**	**3**	**2**	**1**	**3**	**3**	**2**	**1**	**2.2**
CO3	**3**	**2**	**3**	**3**	**3**	**3**	**2**	**1**	**3**	**1**	**2.4**
CO4	**1**	**2**	**1**	**1**	**3**	**2**	**3**	**3**	**1**	**3**	**2.0**
CO5	**3**	**1**	**2**	**1**	**3**	**2**	**3**	**3**	**3**	**2**	**2.3**
CO6											
Mean overall Score											**2.1 (Medium)**

Semester	Course Code	Title of the Course	Hours/ Week	Credits
1	23PCS1CP01	**Core Practical - 1**: Algorithm and OOPS	6	4

Course Objectives
This course covers the basic data structures like Stack, Queue, Tree, List
This course enables the students to learn the application of the data structures using various techniques
It also enables the students to understand C++ language with respect to OOAD concepts
Application of OOPS concepts

List of Programs (75 Hours)

1. Write a program to solve the tower of Hanoi using recursion.
2. Write a program to traverse through binary search tree usingtraversals.
3. Write Program to perform various operations on stack using linked list.
4. Write A Program to perform various operations in a circular queue.
5. Write Program to sort an array an element using quicksort.
6. Write a program to solve number of elements in ascending order using heap sort.
7. Write Program to Solve the knapsack problem using greedy method.
8. Write a program to search for an element in a tree using divide & conquer strategy.
9. Write a program to place the 8 queens on an 8X8 matrix so that no two queens attack.
10. Write a C++ program to perform Virtual Function.
11. Write a C++ program to perform Parameterized constructor.
12. Write a C++ program to perform Friend Function.
13. Write a C++ program to perform Function Overloading.
14. Write a C++ program to perform Single Inheritance.
15. Write a C++ program to perform Employee Details Using files.

Course Outcomes		
CO No.	**CO-Statements** On completion of this course, students will,	**Cognitive Levels** **(K - Level)**
CO1	understand the concepts of object oriented with respect to C++.	**K1**
CO2	be able to understand and implement OOPS concepts	**K2**
CO3	implement data structures like Stack, Queue, Tree, List using C++.	**K3**
CO4	apply data structures for Sorting, Searching using different techniques.	**K4**
CO5	apply and implement major object oriented concepts like function overloading, constructors and inheritance to solve real-world problems.	**K5**
CO6	demonstrate virtual functions and Input/Output Streams.	**K6**

Relationship Matrix

Semester	Course code	Title of the Course	Hours	Credits
1	23PCS1CP01	**Core Practical - 1**: Algorithm and OOPS Lab	6	4

Course Outcomes	Programme Outcomes (POs)					Programme Specific Outcomes (PSOs)					Mean Score of COs
	PO1	PO2	PO3	PO4	PO5	PSO1	PSO2	PSO3	PSO4	PSO5	
CO1	2	3	2	2	2	2	2	3	3	3	2.4
CO2	3	2	2	3	2	2	2	2	3	2	2.3
CO3	3	2	2	3	2	2	2	2	3	2	2.3
CO4	2	2	2	3	2	2	2	3	2	2	2.2
CO5	2	2	3	2	2	2	3	2	2	3	2.3
Mean overall Score 2.34											2.2 (High)

Semester	Course Code	Title of the Course	Hours/ Week	Credits
1	23PCS1ES01	**Elective -1:** Advanced Software Engineering	5	3

Course Objectives
Introduction to Software Engineering, Design, Testing and Maintenance.
Enable the students to learn the concept of Software Engineering.
Learn about Software Project Management, Software Design & Testing.

UNIT I: Introduction (15 Hours)

Introduction: The Problem Domain – Software Engineering Challenges - Software Engineering Approach – Software Processes: Software Process – Characteristics of a Software Process – Software Development Process Models – Other software processes.

UNIT II: Software Requirements (15 Hours)

Software Requirements Analysis and Specification : Requirement engineering – Type of Requirements – Feasibility Studies – Requirements Elicitation – Requirement Analysis – Requirement Documentation – Requirement Validation – Requirement Management – SRS - Formal System Specification – Axiomatic Specification – Algebraic Specification - Case study: Student Result Management System. Software Quality Management –Software Quality, Software Quality Management System, ISO 9000, SEI CMM.

UNIT III: Project Management (15 Hours)

Software Project Management: Responsibilities of a software project manager – Project planning – Metrics for Project size estimation – Project Estimation Techniques – Empirical Estimation Techniques – COCOMO – Halstead‟s software science – Staffing level estimation – Scheduling– Organization and Team Structures – Staffing – Risk management – Software Configuration Management – Miscellaneous Plan.

UNIT IV: Software Design (15 Hours)

Software Design: Outcome of a Design process – Characteristics of a good software design – Cohesion and coupling - Strategy of Design – Function Oriented Design – Object Oriented Design - Detailed Design - IEEE Recommended Practice for Software Design Description.

UNIT V: Software Testing (15 Hours)

Software Testing: A Strategic approach to software testing – Terminologies – Functional testing– Structural testing – Levels of testing – Validation testing - Regression testing – Art of Debugging – Testing tools - Metrics - Reliability Estimation. Software Maintenance - Maintenance Process - Reverse Engineering – Software Re-engineering - Configuration Management Activities.

Teaching Methodology	Videos, PPT, Demonstration and creation of models

Books for Study

1. Jalote, P. (2005). *An integrated approach to software engineering* (3rd ed.). Narosa Publishing House Pvt Ltd, India.
2. Mall, R. (2009). *Fundamentals of software engineering* (3rd ed.). PHI Publication.

Books for Reference

1. Aggarwal, K. K. & Singh, Y.(2008). *Software engineering* (3rd ed.). New Age International Publishers.
2. Pressman, R. S. (2004). *Software engineering: A practitioner's approach* (6th ed.). Published by McGraw Hill.
3. Ghezzi, C., Jarayeri, M. & Manodrioli, D. (2007). *Fundamentals of software engineering (7th ed.). PHI Publication.*

Course Outcomes		
CO No.	**CO-Statements** On completion of this course, students will,	**Cognitive Levels** **(K - Level)**
CO1	understand Software Engineering Process.	**K1**
CO2	understand Software Project Management Skills, design and quality management.	**K2**
CO3	analyze Software Requirements and Specification.	**K3**
CO4	analyze Software Testing, Maintenance and Software Re-Engineering.	**K4**
CO5	design and conduct various types and levels of software quality for software projects.	**K5**
CO6	distinguish Software Testing Strategies.	**K6**

Relationship Matrix

Semester	**Course code**	**Title of the Course**					**Hours**	**Credits**
1	**23PCS1ES01**	**Elective -1:** Advanced Software Engineering					**5**	**3**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Score of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	**3**	**2**	**3**	**2**	**2**	**3**	**2**	**3**	**3**	**3**	**2.6**
CO2	**3**	**2**	**3**	**2**	**1**	**3**	**3**	**2**	**3**	**2**	**2.4**
CO3	**3**	**2**	**1**	**3**	**3**	**2**	**3**	**3**	**2**	**3**	**2.5**
CO4	**2**	**3**	**3**	**2**	**3**	**2**	**2**	**2**	**2**	**3**	**2.4**
CO5	**3**	**2**	**3**	**1**	**3**	**3**	**3**	**3**	**3**	**2**	**2.6**
CO6	**2**	**3**	**3**	**2**	**3**	**2**	**2**	**2**	**2**	**3**	**2.4**
Mean overall Score											**2.4 (High)**

Semester	Course Code	Title of the Course	Hours/ Week	Credits
1	23PCS1ES02	**Elective -2:** Python Programming	5	3

Course Objectives
Present introduction Python, creation web applications, network applications and working in the clouds.
Use functions for structuring Python programs
Understand different Data Structures in Python
Represent compound data using Python lists, tuples and dictionaries

UNIT I: Introduction **(15 Hours)**

Python: Introduction – Numbers – Strings – Variables – Lists – Tuples – Dictionaries – Sets – Comparison.

UNIT II: Code Structures **(15 Hours)**

Code Structures: if, elseif, and else – Repeat with while – Iterate with for – Comprehensions – Functions – Generators – Decorators – Namespaces and Scope – Handle Errors with try and except – User Exceptions.

UNIT III: Modules, Packages, and Programs **(15 Hours)**

Modules, Packages, and Programs: Standalone Programs – Command-Line Arguments – Modules and the import Statement – The Python Standard Library. **Objects and Classes:** Define a Class with class – Inheritance – Override a Method – Add a Method – Get Help from Parent with super – In self Defense –Get and Set Attribute Value with Properties – Name Mangling for Privacy – Method Types – Duck Typing – Special Methods – Composition.

UNIT IV: Data Types **(15 Hours)**

Data Types: Text Strings – Binary Data. **Storing and Retrieving Data:** File Input/Output – Structured Text Files – Structured Binary Files - Relational Databases – NoSQL Data Stores. **Web:** Web Clients – Web Servers – Web Services and Automation.

UNIT V: Systems **(15 Hours)**

Systems: Files – Directories – Programs and Processes – Calendars and Clocks. **Concurrency:** Queues – Processes – Threads – GreenThreads and gevent – twisted– Redis. **Networks:** Patterns – The Publish - Subscribe Model – TCP/IP – Sockets –ZeroMQ – Internet Services – Web Services and APIs – Remote Processing – Big Fat Data and MapReduce – Working in the Clouds.

Teaching Methodology	Videos, PPT, Demonstration and creation of models

Books for Study

1. Lubanovic, B. (2014). *Introducing python* (1st ed.). O'Reilly Inc (Second Release).
2. Lutz, M (2013). *Learning python* (5th ed.). O'Reilly Inc.

Books for Reference

1. Beazley, D. M. (2009). *Python essential edition*. Addison Wesley.
2. Taneja, S. & Naveen, K. (2017). *Python programming - A modular approach* (1st ed.). Pearson India, Pearson Publications.

Course Outcomes		
CO No.	**CO-Statements** On completion of this course, students will,	**Cognitive Levels (K - Level)**
CO1	understand the basic concepts of Python Programming.	**K1**
CO2	understand File Operations, Classes and Objects.	**K2**
CO3	acquire Object Oriented Skills in Python.	**K3**
CO4	develop Web applications using Python.	**K4**
CO5	develop Client Server Networking applications.	**K5**
CO6	discover business applications to solve real time problems.	**K6**

Relationship Matrix											
Semester	**Course code**	**Title of the Course**								**Hours**	**Credits**
1	**23PCS1ES02**	**Elective -2:** Python Programming								**5**	**3**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Score of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	**3**	**2**	**3**	**2**	**2**	**3**	**2**	**3**	**3**	**3**	**2.6**
CO2	**3**	**2**	**3**	**2**	**1**	**3**	**3**	**2**	**3**	**2**	**2.4**
CO3	**3**	**2**	**1**	**3**	**3**	**2**	**3**	**3**	**2**	**3**	**2.5**
CO4	**2**	**3**	**3**	**2**	**3**	**2**	**2**	**2**	**2**	**3**	**2.4**
CO5	**3**	**2**	**3**	**1**	**3**	**3**	**3**	**3**	**3**	**2**	**2.6**
CO6	**3**	**2**	**1**	**3**	**3**	**2**	**3**	**3**	**2**	**3**	**2.5**
Mean overall Score											**2.5 (High)**

Semester	Course Code	Title of the Course	Hours/ Week	Credits
1	23PCS1AE01	**Ability Enhancement Course:** Big Data Analytics	2	1

Course Objectives
Introduction to Big data analytics and Careers in Big data
Understand different Methodologies about Hadoop Technology
This course enables the students to learn the HBase and YARN Technologies

UNIT I: Overview of Big Data (6 Hours)

What is big data – Structuring Big data – Elements of Big data – Big data analytics- Careers in Big data. EXPLORING THE USE OF BIG DATA IN BUSINESS: Use of big data in social networking - Preventing Fraudulent Activities – Detecting Fraudulent Activities in Insurance Sector – Retail Industry.

UNIT II: Technologies for Handling Big Data (6 Hours)

Distributed and parallel computing for Big data – Hadoop – Cloud computing and big data - Understanding Hadoop Ecosystem: Hadoop Ecosystem – Hadoop Distributed File System – Map Reduce.

UNIT III: HBase (6 Hours)

HBase Architecture – Storing big data with HBase – Interacting with the Hadoop Ecosystem – Combining HBase and HDFS – Hive – Pig.

UNIT IV: Big Data Technology (6Hours)

Exploring the big data stack – virtualization and big data. Storing Data in Database and Data Warehouse: RDBMS and Big data.

UNIT V: Hadoop Yarn Architecture (6 Hours)

YARN Architecture – Working of YARN – YARN Schedulers. Exploring Hive: Hive services.

Teaching Methodology	**Videos, PPT, Demonstration and creation of models**

Book for Study

1. DT Editorial Services (2017), *Big data black book*, Dreamtech Press.

Books for Reference

1. Minelli, M., Chambers, M. & Dhiraj, A. (2014). *Big data*. Big Analytics. Wiley.
2. Sathi, A. (2013). *Big Data Analytics: Disruptive technologies for changing the game*. Elsevier.
3. Mohanty, S., Jagadeesh, M. & Srivatsa, H. (2013). *Big data imperatives: Enterprise big data warehouse, BI implementations and analytics*. Apress Media.

Course Outcomes		
CO No.	**CO-Statements** On completion of this course, students will,	**Cognitive Levels (K - Level)**
CO1	perform the fundamentals of various big data analytics techniques.	**K2**
CO2	analyse the HADOOP and Map Reduce technologies associated with Distribution File System	**K5**
CO3	discuss Base Technologies and Hadoop Yarn Architecture	**K6**

Relationship Matrix											
Semester	**Course code**	**Title of the Course**								**Hours**	**Credits**
1	**23PCS1AE01**	**Ability Enhancement Course:** Big Data Analytics								**2**	**1**
Course Outcomes	**Programme Outcomes (POs)**					**Programme Specific Outcomes (PSOs)**					**Mean Score of COs**
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PSO1**	**PSO2**	**PSO3**	**PSO4**	**PSO5**	
CO1	**2**	**3**	**2**	**2**	**2**	**2**	**2**	**3**	**3**	**3**	**2.4**
CO2	**3**	**2**	**2**	**3**	**2**	**2**	**2**	**2**	**3**	**2**	**2.3**
CO3	**2**	**2**	**3**	**2**	**3**	**3**	**2**	**2**	**3**	**3**	**2.5**
Mean overall Score											**2.4 (High)**