

CONTENTS

- i. PO and PSO Description
- ii. PG – Template
- iii. Methods of Evaluation & Methods of Assessment
- iv. Semester Index.
- v. Subjects – Core, Elective, Non-major, Skill Enhanced, Ability Enhanced, Extension Activity, Environment, Professional Competency
 - 1) Course Lesson Box
 - 2) Course Objectives
 - 3) Units
 - 4) Learning Outcome
 - 5) Reference and Text Books
 - 6) Web Sources
 - 7) PO & PSO Mapping tables

TANSICHE REGULATIONS ON LEARNING OUTCOMES-BASED CURRICULUM FRAMEWORK FOR POSTGRADUATE EDUCATION	
Programme	M.Sc., Computer Science
Programme Code	
Duration	PG - Two Years
Programme Outcomes (Pos)	<p>PO1: Problem Solving Skill Apply knowledge of Management theories and Human Resource practices to solve business problems through research in Global context.</p> <p>PO2: Decision Making Skill Foster analytical and critical thinking abilities for data-based decision-making.</p> <p>PO3: Ethical Value Ability to incorporate quality, ethical and legal value-based perspectives to all organizational activities.</p> <p>PO4: Communication Skill Ability to develop communication, managerial and interpersonal skills.</p> <p>PO5: Individual and Team Leadership Skill Capability to lead themselves and the team to achieve organizational goals.</p> <p>PO6: Employability Skill Inculcate contemporary business practices to enhance employability skills in the competitive environment.</p> <p>PO7: Entrepreneurial Skill Equip with skills and competencies to become an entrepreneur.</p> <p>PO8: Contribution to Society Succeed in career endeavors and contribute significantly to society.</p> <p>PO 9 Multicultural competence Possess knowledge of the values and beliefs of multiple cultures and a global perspective.</p> <p>PO 10: Moral and ethical awareness/reasoning Ability to embrace moral/ethical values in conducting one's life.</p>
Programme Specific Outcomes (PSOs)	<p>PSO1 – Placement To prepare the students who will demonstrate respectful engagement with others' ideas, behaviors, beliefs and apply diverse frames of reference to decisions and actions.</p> <p>PSO 2 - Entrepreneur To create effective entrepreneurs by enhancing their critical thinking,</p>

	<p>problem solving, decision making and leadership skill that will facilitate startups and high potential organizations.</p> <p>PSO3 – Research and Development Design and implement HR systems and practices grounded in research that comply with employment laws, leading the organization towards growth and development.</p> <p>PSO4 – Contribution to Business World To produce employable, ethical and innovative professionals to sustain in the dynamic business world.</p> <p>PSO 5 – Contribution to the Society To contribute to the development of the society by collaborating with stakeholders for mutual benefit.</p>
--	---

METHODS OF EVALUATION		
Internal Evaluation	Continuous Internal Assessment Test	25 Marks
	Assignments / Snap Test / Quiz	
	Seminars	
	Attendance and Class Participation	
External Evaluation	End Semester Examination	75 Marks
Total		100 Marks
METHODS OF ASSESSMENT		
Remembering (K1)	<ul style="list-style-type: none"> • The lowest level of questions require students to recall information from the course content • Knowledge questions usually require students to identify information in the textbook. 	
Understanding (K2)	<ul style="list-style-type: none"> • Understanding of facts and ideas by comprehending, organizing, comparing, translating, interpolating and interpreting in their own words. • The questions go beyond simple recall and require students to combine data together 	
Application (K3)	<ul style="list-style-type: none"> • Students have to solve problems by using / applying a concept learned in the class room. • Students must use their knowledge to determine an exact response. 	
Analyze (K4)	<ul style="list-style-type: none"> • Analyzing the question is one that asks the students to break down something in to its component parts. • Analyzing requires students to identify reasons causes or motives and reach conclusions or generalizations. 	
Evaluate (K5)	<ul style="list-style-type: none"> • Evaluation requires an individual to make judgment on something. • Questions to be asked to judge the value of an idea, a character, a work of art, or a solution to a problem. • Students are engaged in decision-making and problem-solving. • Evaluation questions do not have single right answers. 	
Create (K6)	<ul style="list-style-type: none"> • The questions of this category challenge students to get engaged in creative and original thinking. • Developing original ideas and problem solving skills 	

**PROGRAMME OUTCOMES (PO) - PROGRAMME SPECIFIC OUTCOMES (PSO)
MAPPING**

PROGRAMME SPECIFIC OUTCOMES (PSO)					
	PO1	PO2	PO3	PO4	PO5
PSO1	3	3	3	3	3
PSO2	3	3	3	3	3
PSO3	3	3	3	3	3
PSO4	3	3	3	3	3
PSO5	3	3	3	3	3

Level of Correlation between PO's and PSO's

(Suggested by UGC as per Six Sigma Tool – Cause and Effect Matrix)

Assign the value

1 – Low

2 – Medium

3 – High

0 – No Correlation

M.Sc., Computer Science

Course Code	Title of the Course	Credits	Hours		Maximum Marks		
			Theory	Practical	CIA	ESE	Total
FIRSTSEMESTER							
Core - I	Paper I: Analysis & Design of Algorithms	5	7		25	75	100
Core – II	Paper II : Object Oriented Analysis and Design & C++	5	7		25	75	100
Core – III	Paper III : Python Programming	4	6		25	75	100
Elective - I	Paper IV : Advanced Software Engineering / Principles of Compiler Design	3	5		25	75	100
Elective – II	Practical I : Algorithm and OOPS Lab / Python Programming Lab	3	-	5	40	60	100
Total		20	25	5			

I – SEMESTER

Coursecode	ANALYSIS AND DESIGN OF ALGORITHMS			L	T	P	C
Core/Elective/Supportive	Core			7			5
Pre-requisite	Basic Data Structures & Algorithms						
Course Objectives :							
The main objectives of this course are to :							
<ol style="list-style-type: none"> 1. Enable the students to learn the Elementary Data Structures and algorithms. 2. Presents an introduction to the algorithms, their analysis and design. 3. Discuss various methods like Basic Traversal and Search Techniques, Divide and Conquer method, Dynamic programming, backtracking. 4. Understood the various design and analysis of the algorithms. 							
Expected Course Outcomes :							
On the successful completion of the course, student will be able to :							
1	Get knowledge about algorithms and determines their time complexity. Demonstrate specific search and sort algorithms using divide and conquer technique.					K1,K2	
2	Gain good understanding of Greedy method and its algorithm.					K2,K3	
3	Able to describe about graphs using dynamic programming technique.					K3,K4	
4	Demonstrate the concept of backtracking & branch and bound technique.					K5,K6	
5	Explore the traversal and searching technique and apply it for trees and graphs.					K6	
K1-Remember;K2-Understand;K3-Apply;K4-Analyze;K5-Evaluate; K6-Create							
Unit:1	INTRODUCTION					20 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:2	TRAVERSALANDSEARCHTECHNIQUES					20 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:3	GREEDY METHOD					20 hours	
The Greedy Method:-General Method–Knapsack Problem–Minimum Cost Spanning Tree– Single Source Shortest Path.							
Unit:4	DYNAMICPROGRAMMING					20 hours	

Dynamic Programming-General Method–Multi stage Graphs–All Pair Shortest Path–Optimal Binary Search Trees – 0/1 Knapsacks – Traveling Salesman Problem – Flow Shop Scheduling.		
Unit:5	BACKTRACKING	20 hours
Version		
Backtracking:-General Method–8-QueensProblem–Sum of Subsets–Graph Coloring– Hamiltonian Cycles – Branch And Bound: - The Method – Traveling Salesperson.		
Unit:6	Contemporary Issues	5 hours
Expert lectures, online seminars– webinars		
Total Lecture hours		105 hours
Text Books		
1	Ellis Horowitz,“Computer Algorithms”,Galgotia Publications.	
2	Alfred V. Aho, JohnE.Hopcroft, Jeffrey D.Ullman, "Data Structures and Algorithms".	
Reference Books		
1	Goodrich, “Data Structures & Algorithms in Java”, Wiley 3rd edition.	
2	Skiena, ”The Algorithm Design Manual”, Second Edition, Springer, 2008	
3	Anany Levith, ”Introduction to the Design and Analysis of algorithm”, PearsonEducation Asia, 2003.	
4	Robert Sedgewick, Phillipe Flajolet, ” An Introduction to the Analysis of Algorithms”, Addison-Wesley Publishing Company,1996.	
Related Online Contents [MOOC, SWAYAM, NPTEL, Websites etc.,]		
1	https://nptel.ac.in/courses/106/106/106106131/	
2	https://www.tutorialspoint.com/design_and_analysis_of_algorithms/index.htm	
3	https://www.javatpoint.com/daa-tutorial	

Mapping with Programming Outcomes										
Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	M	S	M	S	L	M	L	S	M
CO2	S	S	S	S	S	M	S	M	S	M
CO3	S	S	S	S	S	M	S	M	S	M
CO4	S	S	S	S	S	M	S	M	S	M
CO5	S	S	S	S	S	M	S	M	S	M

*S-Strong; M-Medium; L-Low

I – SEMESTER

Course code	OBJECT ORIENTED ANALYSIS AND DESIGN & C++	L	T	P	C
Core/Elective/Supportive	Core	7			5
Pre-requisite	Basics of C++ and Object Oriented Concepts				
Course Objectives :					
The main objectives of this course are to :					
<ol style="list-style-type: none"> 1. Present the object model, classes and objects, object orientation, machine view and model management view. 2. Enables the students to learn the basic functions, principles and concepts of object oriented analysis and design. 3. Enable the students to understand C++ language with respect to OOAD 					
Expected Course Outcomes :					
On the successful completion of the course, student will be able to :					
1	Understand the concept of Object-Oriented development and modeling techniques				K1,K2
2	Gain knowledge about the various steps performed during object design				K2,K3
3	Abstract object-based views for generic software systems				K3
4	Link OOAD with C++ language				K4,K5
5	Apply the basic concept of OOPs and familiarize to write C++ program				K5,K6
K1-Remember;K2-Understand;K3-Apply;K4-Analyze;K5-Evaluate; K6-Create					
Unit:1	OBJECT MODEL	20 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:2	CLASSES AND OBJECTS	20 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:3	C++INTRODUCTION	20 hours			
Introduction to C++ - Input and output statements in C++ - Declarations – Control Structures – Functions in C++.					
Unit:4	INHERITANCE AND OVER LOADING	20 hours			

Classes and Objects – Constructors and Destructors – operators overloading – Type Conversion - Inheritance – Pointers and Arrays.

Unit:5	POLYMORPHISM AND FILES	20 hours
Memory Management Operators – Polymorphism – Virtual functions – Files – Exception Handling – String Handling - Templates.		
Unit:6	Contemporary Issues	5 hours
Expert lectures, online seminars – webinars		
Total Lecture hours		105 hours

Text Books

1	“Object Oriented Analysis and Design with Applications”, Grady Booch, Second Edition, Pearson Education.
2	“Object-Oriented Programming with ANSI & TurboC++”, Ashok N. Kamthane, First Indian Print -2003, Pearson Education.

Reference Books

1	Balagurusamy, “Object Oriented Programming with C++”, TMH, Second Edition, 2003.
---	--

Related Online Contents [MOOC, SWAYAM, NPTEL, Websites etc.,]

1	https://onlinecourses.nptel.ac.in/noc19_cs48/preview
2	https://nptel.ac.in/noc/courses/noc16/SEM2/noc16-cs19/
3	https://www.tutorialspoint.com/object_oriented_analysis_design/oad_object_oriented_analysis.htm

Mapping with Programming Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	S	S	M	S	M	S	M	S	S
CO2	S	S	S	M	S	M	S	M	S	S
CO3	S	S	S	M	S	M	S	M	S	S
CO4	S	S	S	M	S	M	S	M	S	S
CO5	S	S	S	M	S	M	S	M	S	S

*S-Strong; M-Medium; L-Low

I – SEMESTER

Course code	PYTHON PROGRAMMING			L	T	P	C
Core/Elective/Supportive	Core			6			4
Pre-requisite	Basics of any OO Programming Language						
Course Objectives :							
The main objectives of this course are to :							
<ol style="list-style-type: none"> 1. Presents an introduction to Python, creation of web applications, network applications and working in the clouds 2. Use functions for structuring Python programs 3. Understand different Data Structures of Python 4. Represent compound data using Python lists, tuples and dictionaries 							
Expected Course Outcomes :							
On the successful completion of the course, student will be able to :							
1	Understand the basic concepts of Python Programming					K1,K2	
2	Understand File operations, Classes and Objects					K2,K3	
3	Acquire Object Oriented Skills in Python					K3,K4	
4	Develop web applications using Python					K5	
5	Develop Client Server Networking applications					K5,K6	
K1-Remember;K2-Understand;K3-Apply;K4-Analyze;K5-Evaluate; K6-Create							
Unit:1	INTRODUCTION					18 hours	
Python: Introduction–Numbers–Strings–Variables–Lists–Tuples–Dictionaries–Sets–Comparison.							
Unit:2	CODE STRUCTURES					18 hours	
Code Structures: if, else if, and else – Repeat with while – Iterate with for – Comprehensions – Functions – Generators – Decorators – Namespaces and Scope – Handle Errors with try and except – User Exceptions.							
Unit:3	MODULES, PACKAGES AND CLASSES					18 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–Inself Defense –Get and Set Attribute Values with Properties –Name Mangling for Privacy – Method Types – Duck Typing – Special Methods –Composition.							
Unit:4	DATATYPESANDWEB					18 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:5	SYSTEMS AND NETWORKS	15 hours
Systems : Files–Directories–Programs and Processes–Calendars and Clocks.		
Concurrency: Queues– Processes–Threads–Green Threads 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.		
Unit:6	Contemporary Issues	3 hours
Expert lectures, online seminars – webinars		
	Total Lecture hours	90 hours
Text Books		
1	Bill Lubanovic, “Introducing Python”, O’Reilly, First Edition – Second Release, 2014.	
2	Mark Lutz, “Learning Python”, O’Reilly, Fifth Edition, 2013.	
Reference Books		
1	David M. Beazley, “Python Essential Reference”, Developer’s Library, Fourth Edition, 2009.	
2	Sheetal Taneja, Naveen Kumar, “Python Programming-A Modular Approach”, Pearson Publications.	
Related Online Contents [MOOC, SWAYAM, NPTEL, Websites etc.,]		
1	https://www.programiz.com/python-programming/	
2	https://www.tutorialspoint.com/python/index.htm	
3	https://onlinecourses.swayam2.ac.in/aic20_sp33/preview	

Mapping with Programming Outcomes										
Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	S	M	S	S	S	M	M	S	M
CO2	S	S	S	S	S	S	S	M	S	M
CO3	S	S	S	S	S	S	S	M	S	M
CO4	S	S	S	S	S	S	S	M	S	M
CO5	S	S	S	S	S	S	S	M	S	M

*S-Strong; M-Medium; L-Low

I – SEMESTER

Course code	ADVANCED SOFTWARE ENGINEERING	L	T	P	C
Core/Elective/Supportive	Elective	5			3
Pre-requisite	Basics of Software Engineering & SPM				
Course Objectives :					
The main objectives of this course are to :					
<ol style="list-style-type: none"> 1. Introduce to Software Engineering, Design, Testing and Maintenance. 2. Enable the students to learn the concepts of Software Engineering. 3. Learn about Software Project Management, Software Design & Testing. 					
Expected Course Outcomes :					
On the successful completion of the course, student will be able to :					
1	Understand about Software Engineering process				K1,K2
2	Understand about Software project management skills, design and quality management				K2,K3
3	Analyze on Software Requirements and Specification				K3,K4
4	Analyze on Software Testing, Maintenance and Software Re-Engineering				K4,K5
5	Design and conduct various types and levels of software quality for a software project				K5,K6
K1-Remember;K2-Understand;K3-Apply;K4-Analyze;K5-Evaluate; K6-Create					
Unit:1	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:2	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:3	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:4	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 Descriptions.		
Unit:5	SOFTWARE TESTING	13 hours
Software Testing: A Strategic approach to software testing – Terminologies – Functional testing– Structural testing – Levels of testing – Validation testing - Regression testing – Art of Debugging–Testingtools-Metrics-ReliabilityEstimation.SoftwareMaintenance -Maintenance Process - Reverse Engineering – Software Re-engineering - Configuration Management Activities.		
Unit:6	Contemporary Issues	2 hours
Expert lectures, online seminars – webinars		
	Total Lecture hours	75 hours
Text Books		
1	An Integrated Approach to Software Engineering – Pankaj Jalote, Narosa Publishing House, Delhi, 3rd Edition.	
2	Fundamentals of Software Engineering – Rajib Mall, PHI Publication, 3 rd Edition.	
Reference Books		
1	Software Engineering – K.K.Aggarwal andYogesh Singh, New Age International Publishers, 3rd edition.	
2	A Practitioners Approach – Software Engineering, R.S. Pressman, McGraw Hill.	
3	Fundamentals of Software Engineering - Carlo Ghezzi, M. Jarayeri, D. Manodrioli, PHI Publication.	
Related Online Contents [MOOC, SWAYAM, NPTEL, Websites etc.,]		
1	https://www.javatpoint.com/software-engineering-tutorial	
2	https://onlinecourses.swayam2.ac.in/cec20_cs07/preview	
3	https://onlinecourses.nptel.ac.in/noc19_cs69/preview	

Mapping with Programming Outcomes										
Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	S	M	S	S	S	M	M	M	M
CO2	S	S	S	S	S	S	S	M	S	S
CO3	S	S	S	S	S	S	S	M	S	S
CO4	S	S	S	S	S	S	S	M	S	S
CO5	S	S	S	S	S	S	S	M	S	S

*S-Strong; M-Medium; L-Low

I – SEMESTER

Course code	PRINCIPLES OF COMPILER DESIGN	L	T	P	C
Core/Elective/Supportive	Elective	5			3
Pre-requisite	Basics of Formal Languages and Automata Theory				
Course Objectives :					
The main objectives of this course are to :					
<ol style="list-style-type: none"> 1. Introducing Grammar, Finite Automata, Parser, Syntax Tree and Code Generation. 2. Enable the students to learn about different phases of Compiler. 3. Learn about Conversion of Source Code to Object Code. 					
Expected Course Outcomes :					
On the successful completion of the course, student will be able to :					
1	Understand the phases and tools available in Compiler				K2
2	Design and implement a Lexical Analyzer				K3
3	Compare and analyze different types of Compilers				K4
4	Specify appropriate translations to generate Intermediate Code				K3
5	Identify sources for Code Optimization				K4
K1-Remember;K2-Understand;K3-Apply;K4-Analyze;K5-Evaluate; K6-Create					
Unit:1	INTRODUCTION TO COMPILERS	15 hours			
Introduction to Compiling – Compilers – Analysis of the Source Program – Phases of a Compiler. Grouping of Phases – Compiler Construction Tools.					
Unit:2	LEXICAL ANALYSIS	15 hours			
Lexical Analysis – Role of the Lexical Analyzer – Specification and Recognition of Tokens – Language for specifying Lexical Analyzer – Finite Automata – Regular Expressions to NFA – Design of Lexical Analyzer Generator – Optimization of DFA based pattern matchers.					
Unit:3	SYNTAX ANALYSIS	15 hours			
Syntax Analysis – Role of Parser – Context Free Grammars – Top Down Parsing – Bottom Up Parsing – Operator Precedence Parsing – LR Parsers.					
Unit:4	SYNTAX DIRECTED TRANSLATION	15 hours			

Syntax Directed Translation: Syntax Directed Definitions – Construction of Syntax Trees – Bottom Up evaluation of attributed definition – Bottom Up evaluation of inherited attributes – Recursive evaluators.		
Unit:5	INTERMEDIATE CODE GENERATION AND OPTIMIZATION	12 hours
Intermediate Code Generation: Intermediate Languages – Declaration – Assignment Statements. Procedure Calls – Runtime Storage Management. Code Generation and Optimization: Basic Blocks and Flow Graphs – DAG Representation.		
Unit:6	Contemporary Issues	3 hours
Expert lectures, online seminars – webinars		
Total Lecture hours		75 hours
Text Books		
1	Compilers – Principles, Techniques and Tools – Alfred Aho, Ravi Sethi, Jeffrey D. Ullman, Pearson : 1986	
2	Modern Compiler Design – Dick Grune, Bal, Langendoen, Jacobs, Wiley : 2012	
3	Compiler Design – K. Muneeswaran, Oxford University Press : 2013	
Reference Books		
1	Modern Compiler Design – David Galles, Pearson Education Asia : 2001	
2	Advanced Compiler Design and Implementation – Steven S. Muchnick, Morgan Kaufmann Publishers : 2000	
3	Crafting a Compiler with C – C.N.Fisher, R.J. Le Blane, Pearson Education : 2000	
Related Online Contents [MOOC, SWAYAM, NPTEL, Websites etc.,]		
1	https://www.geeksforgeeks.org/compiler-lexical-analysis	
2	https://ieeexplore.ieee.org/document/7779385/	
3	https://www.tutorialspoint.com/compiler_design/compiler_design_tutorial.pdf	

Mapping with Programming Outcomes										
Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	M	M	S	S	M	-	-	-	-
CO2	S	S	S	M	M	S	-	-	-	-
CO3	S	S	S	S	S	M	-	-	-	-
CO4	S	S	S	S	M	M	-	-	-	-
CO5	S	S	S	M	S	S	-	-	-	-

*S-Strong; M-Medium; L-Low

I – SEMESTER

Course code	PRACTICAL I : ALGORITHM AND OOPS LAB	L	T	P	C
Core/Elective/Supportive	Core			5	3
Pre-requisite	Basic Programming of C++ language				
Course Objectives:					
The main objectives of this course are to :					
<ol style="list-style-type: none"> 1. This course covers the basic data structures like Stack, Queue, Tree, List. 2. This course enables the students to learn the applications of the data structures using various techniques 3. It also enable the students to understand C++ language with respect to OOAD concepts 4. Application of OOPS concepts. 					
Expected Course Outcomes :					
On the successful completion of the course, student will be able to :					
1	Understand the concepts of object oriented with respect to C++				K1,K2
2	Able to understand and implement OOPS concepts				K3,K4
3	Implementation of data structures like Stack, Queue, Tree, List using C++				K4,K5
4	Application of the data structures for Sorting, Searching using different techniques.				K5,K6
K1-Remember;K2-Understand;K3-Apply;K4-Analyze;K5-Evaluate; K6-Create					
LISTOF PROGRAMS					75 hours
<ol style="list-style-type: none"> 1) Write a program to solve the tower of Hanoi using recursion. 2) Write a program to traverse through binary search tree using traversals. 3) Write a program to perform various operations on stack using linked list. 4) Write a program to perform various operations in circular queue. 5) Write a program to sort an array of an elements using quick sort. 6) Write a program to solve number of elements in ascending order using heap sort. 7) Write a 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 8 X 8 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. 					
Total Practical hours					75 hours

Text Books	
1	Goodrich, "Data Structures & Algorithms in Java", Wiley 3rd edition.
2	Skiena, "The Algorithm Design Manual", Second Edition, Springer, 2008
Reference Books	
1	Anany Levith, "Introduction to the Design and Analysis of algorithm", Pearson Education Asia, 2003.
2	Robert Sedgewick, Phillipe Flajolet, "An Introduction to the Analysis of Algorithms", Addison-Wesley Publishing Company, 1996.
Related Online Contents[MOOC, SWAYAM, NPTEL, Websites etc.,]	
1	https://onlinecourses.nptel.ac.in/noc19_cs48/preview
2	https://nptel.ac.in/noc/courses/noc16/SEM2/noc16-cs19/
3	https://www.tutorialspoint.com/object_oriented_analysis_design/ooad_object_oriented_analysis.htm

Mapping with Programming Outcomes										
COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	S	M	S	S	S	M	M	S	S
CO2	S	S	S	S	S	S	S	M	S	S
CO3	S	S	S	S	S	S	S	M	S	S
CO4	S	S	S	S	S	S	S	M	S	S

*S-Strong; M-Medium; L-Low

I – SEMESTER

Course code		PRACTICAL I : PYTHON PROGRAMMING LAB	L	T	P	C
Core/Elective/Supportive		Core			5	3
Pre-requisite	Basics of any OO Programming Language					
Course Objectives:						
The main objectives of this course are to :						
<ol style="list-style-type: none"> 1. This course presents an overview of elementary data items, lists, dictionaries, sets and tuples 2. To understand and write simple Python programs 3. To Understand the OOPS concepts of Python 4. To develop web applications using Python 						
Expected Course Outcomes :						
On the successful completion of the course, student will be able to :						
1	Able to write programs in Python using OOPS concepts				K1,K2	
2	To understand the concepts of File operations and Modules in Python				K2,K3	
3	Implementation of lists, dictionaries, sets and tuples as programs				K3,K4	
4	To develop web applications using Python				K5,K6	
K1-Remember;K2-Understand;K3-Apply;K4-Analyze;K5-Evaluate; K6-Create						
LISTOF PROGRAMS					75 hours	
Implement the following in Python :						
<ol style="list-style-type: none"> 1. Programs using elementary data items, lists, dictionaries and tuples 2. Programs using conditional branches, 3. Programs using loops. 4. Programs using functions 5. Programs using exception handling 6. Programs using inheritance 7. Programs using Polymorphism 8. Programs to implement file operations. 9. Programs using modules. 10. Programs for creating dynamic and interactive web pages using forms. 						
Total Practical hours					75 hours	
Text Books						
1	Bill Lubanovic, “Introducing Python”,O’Reilly, First Edition - SecondRelease,2014.					
2	MarkLutz, “LearningPython”, O’Reilly,FifthEdition, 2013.					
Reference Books						

1	David M. Beazley, "Python Essential Reference", Developer's Library, Fourth Edition, 2009.
2	Sheetal Taneja, Naveen Kumar, "Python Programming - A Modular Approach", Pearson Publications.
Related Online Contents [MOOC, SWAYAM, NPTEL, Websites etc.,]	
1	https://www.programiz.com/python-programming/
2	https://www.tutorialspoint.com/python/index.htm
3	https://onlinecourses.swyam2.ac.in/aic20_sp33/preview

Mapping with Programming Outcomes										
COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1	S	S	M	S	S	S	M	M	S	S
CO2	S	S	S	S	S	S	S	M	S	M
CO3	S	S	S	S	S	S	S	M	S	S
CO4	S	S	S	S	S	S	S	M	S	S

*S-Strong; M-Medium; L-Low