

Course Code CSE1021	Introduction to Problem Solving and Programming	Course Type	LTP
		Credits	4
Pre-requisite : Nil			
Course Objectives:			
<ul style="list-style-type: none">• The course aims to provide exposure to problem-solving through programming• It aims to train the student to the basic concepts of the programming in any language.• Introduce the essential skills for a logical thinking for problem solving solving• Gain knowledge about the different data types and control flow statements• Impart knowledge about the functions, list, set tuples and dictionaries• Learn Scratch and Raptor tools for programming and flowchart.			
Course Outcomes:			
Students will be able to solve problems, explore real-world Computational Engineering challenges, and create practical and contemporary applications. At the completion of this course, students should be able to do the following:			
<ul style="list-style-type: none">• Apply the algorithm design techniques to analyze, solve and evaluate computing problems.• Identify situations where computational methods and computers would be useful.• Given a computational problem, identify and abstractthe programming task involved.• Approach the programming tasks using techniques learned and write pseudo-code.• Identify tasks in which the numerical techniques learned are applicable and apply them to write programs in python language, and hence use computers effectively to solve the task.• Read, write, execute by hand simple Python programs.• Represent compound data using Python lists, tuples, dictionaries			
Unit No	Unit Content	No. of hours	
1	Introduction to Computer Problem Solving: Introduction – Problem Solving Aspect – Top-Down Design – Implementation of Algorithms (flowcharts, pseudo code, programming language) – Program Verification – Efficiency of Algorithms – Analysis of Algorithms.	08	
2	Python Data, Expressions and Statements: Python interpreter and interactive mode – values and types – variables, expressions, statements, tuple assignment, precedence of operators, comments – modules and functions, function definition and use, flow of execution, parameters and arguments.	08	
3	Fundamental Algorithms: Introduction – Exchange the values – Counting – Summation – Factorial Computation – Fibonacci Sequence – Reverse – Base Conversion – Character to Number Conversion Python Control Flow, Functions : Conditionals: Boolean values and operators, conditional (if, if- else, if-elif-else) – Iteration statements (state, while, for break, continue, pass);	11	
4	Factoring Methods: Finding square root – Smallest Divisor – GCD – Generate Prime Numbers – Computing Prime Factors – Generating Pseudo-random numbers – Raising a Number to Large Power – Computing n th Fibonacci Numbers	09	
5	Array Techniques: Introduction – Array Order Reversal – Array Counting – Finding the Maximum number in a set – Removal of Duplicates from an ordered array – Partitioning an array – Finding the K th smallest element Python Lists: list operations – Tuples – Sets Operations – Dictionaries – Time Tradeoff.	09	
6	Tools Required : Scratch, Raptor		
	Total Hours:	45	

Mode of Teaching and Learning: <i>Flipped Class Room, Activity Based Teaching/Learning, Digital/Computer based models, wherever possible to use scratch and raptor tools</i>	
Mode of Evaluation and assessment: <i>The assessment and evaluation components may consist of unannounced open book examinations, quizzes, student's portfolio generation and assessment, and any other innovative assessment practices followed by faculty, in addition to the Continuous Assessment Tests and Final Examinations.</i>	
Text Books:	
1.	Dromey R G, "How to Solve it by Computer", PHI
2	C. Dierbach, Introduction to Computer Science Using PYTHON: A Computational Problem-Solving Focus (1 ed.), Wiley, 2015. ISBN 978-0470555156.
3	A.R. Bradley, "Programming for Engineers", Springer, 2011
4	Yashavant Kanetkar, Let Us Python (1 ed.), BPB Publishers, 2019. ISBN 978-9388511568.
Reference Books:	
1.	G. Brassard and P. Bratley - Fundamentals of Algorithmics - PHI - 2001.
2	Levitin, "Introduction to the Design and Analysis of Algorithms", 3 rd , Pearson, 2011
3.	Allen B. Downey, Think Python: How to Think Like a Computer Scientist (2 ed.), O'Reilly, 2015. ISBN 978-9352134755.
4.	Martin C. Brown, Python: The Complete Reference (1 ed.), McGraw-Hill, 2001. ISBN 978-9387572942.

INDICATIVE LIST OF EXPERIMENTS

No.	Description of Experiment
1	Problem Solving: Drawing flowchart using yEd tool/Raptor Tool
2	Basic programs in Python : Input and output Operations
3	Programs using basic commands and simple conditional statements
4	Programs using conditional statements and looping
5	Programs using Factoring Methods
6	Programs using Array Technique
7	Programs using Lists

Recommendation by the Board of Studies on	-
Approval by Academic council on	-
Compiled by	Dr. Muneeswaran V.