PROBLEM SOLVING AND PROGRAMMING WITH PYTHON

Course Code	20ES1102	Year	I	Semester	I
Course Category	Engineering Science	Branch	ME	Course Type	Theory
Credits	3	L-T-P	3-0-0	Pre-requisites	Nil
Continuous Internal Evaluation	30	Semester End Evaluation	70	Total Marks	100

Course Outcomes: Upon successful completion of the course, the student will be able to

CO	Statement	Skill	BTL	Units
CO1	Understand the principles of structured programming and C constructs for solving problems.	Understand	L2	1,2,3,4,5
CO2	Apply suitable control constructs and array concepts to solve problems.	Apply	L3	1,2,
CO3	Apply the concept of pointers, user defined data types and files to solve problems.	Apply	L3	3,4,5
CO4	Analyze the given problem and use modular programming approach to develop solutions.	Analyze	L4	5

Contribution of Course Outcomes towards achievement of Program Outcomes & Strength of correlations (3: High, 2: Medium, 1: Low)

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3												2	
CO2	3												2	
CO3	3								3	3			2	
CO4		2											2	

Syllabus

UNIT	Contents	Mapped COs
I	**Computational Thinking and Visual Programming Concepts** Introduction to computational thinking. Visual programming concepts. Scratch environment: sprites -- appearance and motion, angles and directions, repetition and variation, changing costumes, adding background, Input/output, variables and operators. **Problems** - draw geometrical shapes such as Circle, Triangle, Square and Pentagon, Make a sprite to ask the user to enter two different numbers and an arithmetic operator and then calculate and display the result, make a sprite to ask the user to enter a number to display even and odd numbers.	**CO1 CO2**
II	**Algorithms and Flowchart design through Raptor** Introduction to the idea of an algorithm, Pseudo code and Flowcharts. Flowchart symbols, Input/output, Assignment, operators, conditional if, repetition, procedure and sub charts. **Problems** - Finding maximum of 3 numbers, Unit converters, Interest calculators, and multiplication tables, GCD of 2 numbers,	**CO1 CO2**

	Fibonacci number generation, and prime number generation. Minimum, Maximum and average of n numbers.	
III	**Introduction to Python** Features of Python, Writing and Executing First Python Program, Literal Constants, Variables and Identifiers, Reserved Words, Data Types, Input Operation, Operators and Expressions, Operations on Strings, Type Conversion, Conditional statements and iterative statements.	**CO1** **CO3**
IV	**Functions and Strings in Python** **Functions:** Introduction, Built-in Math Functions, User Defined Functions: Function Call, Variable Scope and Lifetime, The return statement, Lambda Functions, Packages in python. **Strings:** Introduction, Built-in String Functions, Slice Operation, Comparing Strings, Iterating String, Regular Expressions.	**CO1** **CO3**
V	**Files and Data Structures in Python** **File Handling:** open, close, read and write operations. **Data Structures:** **Lists:** Accessing values in lists, Nested Lists, Basic List Operations. **Tuples:** Creating Tuple, Accessing values in a tuple, Basic Tuple Operations. **Dictionaries**: Creating and Accessing Dictionaries, Built-in Dictionary functions, List Vs Tuple Vs Dictionary.	**CO1** **CO3** **CO4**

Learning Resources
Text Books
1. An introduction to programming and algorithmic reasoning using raptor, Weingart, Dr.Troy, Wayne, 2018, CreateSpace (an Amazon.com Company) 2. Core Python Programming, R. Nageswara Rao, 2018, Dreamtech press.
Reference Books
1. Python Programming: Using Problem Solving Approach, Reema Thareja, 2017, Oxford University Press. 2. Programming with python, T R Padmanabhan, 2017, Springer. 3. Python for Data Analysis, Wes McKinney, 2012, O.Reilly.
e- Resources & other digital material
1. http://fusecontent.education.vic.gov.au/9f79537a-66fc-4070-a5ce-e3aa315888a1/scratchreferenceguide14.pdf 2. https://raptor.martincarlisle.com/ 3. http://www.ict.ru.ac.za/Resources/cspw/thinkcspy3/thinkcspy3.pdf