

Department of CSE, PES University

UE23CS151A

Python for Computational Problem Solving (5-0-2-5-5)

Python is an easy to learn, general-purpose, powerful programming language. It has efficient high-level data structures and a simple but effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, together with its interpreted nature, make it an ideal language for scripting and rapid application development in many areas on most platforms.

Course Objectives:

- Learn the syntax and semantics of Python programming language.
- Illustrate the process of structuring the data using lists, tuples, sets and dictionaries.
- Demonstrate the use of built-in functions to navigate the file system.
- Learn various paradigms of programming and implement the Object Oriented Programming concepts in Python.

Course Outcomes:

At the end of this course students will be able to,

- Program effectively using the Python language.
- Identify the methods to create and manipulate lists, tuples and dictionaries.
- Discover commonly used operations involving file system.
- Think using different paradigms of programming and interpret the concepts of Object-Oriented Programming as used in Python.

Course Contents:

Unit 1: Introduction

Computation Problem Solving-Limits of Computational Problem Solving - Computer Algorithm – Computer Hardware. Digital Computer - Operating System- Limits of IC technology - Computer Software - Syntax, semantics and program translation, Introduction to Python Programming Language, IDLE Python Development Environment, Output function - variables, types and id, input function, operators and expressions, Control structures.

12 Hours

Unit 2: Collections & Functions

Lists, Tuples, Dictionaries, Sets, Strings and text file manipulation: reading and writing files. Functions: Definition, call Positional and keyword parameter, Default parameters, Variable number of arguments.

16 Hours

Unit 3: Functions, GUI, Modules, Testing and Debugging

Recursion, Call-backs, Closure, Decorators, generators. Graphical User Interface with Tkinter package- Different geometric methods – Tk, mainloop, Creating simple GUI - buttons, canvas, check button, labels, entry fields, dialogs Widgets - sizes, fonts, colours layouts, nested frames, Modules - import mechanisms. Testing- Pytest , Function testing with Doctest, pdb debugger commands.

14 Hours

Unit 4: Functional & Object Oriented Programming

Lambda function, Map, filter, and reduce, max, min, Zip, list comprehension.

Classes and objects - inheritance, polymorphism, iterators, Error handling & Exceptions - try, except and raise, exception propagation.

14 Hours

Tools / Languages: Python.

Text Book(s):

1: “Introduction to Computer Science Using Python: A Computational Problem- Focus”, Charles Dierbach, Wiley India Edition, John Wiley, 2015

Reference Book(s):

- 1: “Learn python Programming”, Fabrizio Romano, 2nd Edition, Packet Publishing, 2018.
- 2: “Fundamentals of Python: First Programs”, Kenneth A. Lambert, Cengage, 2019.
- 3: “Introduction to Computation and Programming Using Python: With Application to Understanding Data”, John V. Guttag, MIT Press, MIT with Library of Congress Cataloguing- in-Publication Data, 2016.

Lab / Hands-on:

- 1: Programs on Input Output Functions, Operators and Expressions, Usage of Libraries and Control Structures.
- 2: Programs on Collections (Lists, Tuples, Sets , Dictionaries , Strings).
- 3: Programs on Files and File manipulations.
- 4: Programs on Functions.
- 5: Programs on Functional Programming.
- 6: Programs on Object Oriented Programming.