



Department of Computer Science and Engineering

PES University, Bangalore, India

Python For Computational Problem Solving(4-0-2-5-5)

UE24CS151A

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

14 Hours

Computational 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.

Unit 2: Collections & Functions

14 Hours

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.

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

14 Hours

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.

Unit 4: Functional & Object Oriented Programming

14 Hours

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.

Tools / Languages: Python interpreter 3.8 and above. IDLE and any IDE like Jupyter

Text Book(s):

1. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist", 2 nd Edition, Green Tea Press, 2015. (Available under CC-BY-NC license at <http://greenteapress.com/thinkpython2/thinkpython2.pdf> (Chapters 13, 15, 16, 17, 18) (Download pdf/html files from the above link)
2. Al Sweigart, "Automate the Boring Stuff with Python", 1 st Edition, No Starch Press, 2015. (Available under CC-BY-NC-SA license at <https://automatetheboringstuff.com/>)
3. Reema Thareja, Python Programming using Problem Solving approach, Second Edition.

Reference Book(s):

- 1: "Introduction to Computer Science Using Python: A Computational Problem- Focus", Charles Dierbach, Wiley India Edition, John Wiley, 2015.
2. "Learn python Programming", Fabrizio Romano, 2nd Edition, Packet Publishing, 2018.
3. "Fundamentals of Python: First Programs", Kenneth A. Lambert, Cengage, 2019.
4. "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:

- Programs on Input Output Functions, Operators and Expressions and Usage of Libraries
- Programs on Control Structures.
- Programs on Collections (Lists, Tuples, Sets , Dictionaries , Strings).
- Programs on Files and File manipulations.
- Programs on Functions.
- Programs on Functional Programming.
- Programs on Object Oriented Programming.
- Programs on Pytest, pdb and GUI