**19CS41O1 - ADVANCED PYTHON PROGRAMMING**

|  |  |  |  |
| --- | --- | --- | --- |
| **Course Category:** | Open Elective | **Credits:** | 3 |
| **Course Type:** | Theory | **Lecture-Tutorial-Practical:** | 3-0-0 |
| **Pre-requisite:** | Basic mathematical knowledge to solve problems and programming. | **Sessional Evaluation: Univ. Exam Evaluation:**  **Total Marks:** | 40  60  100 |

|  |  |  |
| --- | --- | --- |
| **Course**  **Objectives:** | Students undergoing this course are expected to learn:   * To know the basics of algorithmic problem solving * To read and write simple Python programs. * To develop Python programs with conditionals and loops. * To define Python functions and call them. * To use Python data structures – lists, tuples, dictionaries. * To do input/output with files in Python. | |
| **Course Outcomes** | Upon successful completion of the course, the students will be able to: | |
| CO1 | Develop algorithmic solutions to simple computational problems |
| CO2 | Read, write, execute by hand simple Python programs. |
| CO3 | Structure simple Python programs for solving problems. |
| CO4 | Decompose a Python program into functions. |
| CO5 | Represent compound data using Python lists, tuples, dictionaries. |
| CO6 | Read and write data from/to files in Python Programs. |
| **Course Content** | **UNIT-I**  **ALGORITHMIC PROBLEM SOLVING**  Algorithms, building blocks of algorithms (statements, state, control flow, functions), notation (pseudo code, flow chart, programming language), algorithmic problem solving, simple strategies for developing algorithms (iteration, recursion). Illustrative problems: find minimum in a list, insert a card in a list of sorted cards, guess an integer number in a range, Towers of Hano.  **UNIT-II DATA, EXPRESSIONS, STATEMENTS**  Python interpreter and interactive mode; values and types: int, float, boolean, string, and list; variables, expressions, statements, tuple assignment, precedence of operators, comments; modules and functions, function definition and use, flow of execution, parameters and arguments; Illustrative programs: exchange the values of two variables, circulate the values of n variables, distance between two points.  **UNIT-III**  **CONTROL FLOW**  Conditionals: Boolean values and operators, conditional (if), alternative (if-else), chained conditional (if-elif-else); Iteration: state, while, for, break, continue, pass;  **UNIT-IV**  **FUNCTIONS**  Fruitful functions: return values, parameters, local and global scope, function composition, recursion; Strings: string slices, immutability, string functions and methods, string module; Lists as arrays. Illustrative programs: square root, gcd, exponentiation, sum an array of numbers, linear search, binary search.  **UNIT-V**  **LISTS, TUPLES, DICTIONARIES**  Lists: list operations, list slices, list methods, list loop, mutability, aliasing, cloning lists, list parameters; Tuples: tuple assignment, tuple as return value; Dictionaries: operations and methods; advanced list processing - list comprehension; Illustrative programs: selection sort, insertion sort, merge sort, histogram.  **UNIT-VI**  **FILES, MODULES, PACKAGES**  Files and exception: text files, reading and writing files, format operator; command line arguments, errors and exceptions, handling exceptions, modules, packages; Illustrative programs: word count, copy file. | |
| **Text Books**  **&**  **References Books** | **TEXT BOOKS**   1. Allen B. Downey, ``Think Python: How to Think Like a Computer Scientist’’, 2nd edition, Updated for Python 3, Shroff/O’Reilly Publishers, 2016 [(http://gre](http://greenteapress.com/wp/think-)e[nteapress.com/wp/think-](http://greenteapress.com/wp/think-) python/) 2. Guido van Rossum and Fred L. Drake Jr, “An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd.,2011.   **REFERENCE BOOKS**   1. Charles Dierbach, “Introduction to Computer Science using Python: A Computational Problem-Solving Focus, Wiley India Edition,2013. 2. John V Guttag, “Introduction to Computation and Programming Using Python’’, Revised and expanded Edition, MIT Press ,2013 3. Kenneth A. Lambert, “Fundamentals of Python: First Programs”, CENGAGE Learning, 2012. 4. Paul Gries, Jennifer Campbell and Jason Montojo, “Practical Programming: An Introduction to Computer Science using Python 3”, Second edition, Pragmatic Programmers, LLC, 2013. 5. Robert Sedgewick, Kevin Wayne, Robert Dondero, “Introduction to Programming in Python: An Inter-disciplinary Approach, Pearson India Education Services Pvt. Ltd.,2016.   6.Timothy A. Budd, “Exploring Python”, Mc-Graw Hill Education (India) Private  Ltd., 2015. | |
| **E-Resources** | 1. <https://nptel.ac.in/courses> 2. <https://freevideolectures.com/university/iitm> 3. <https://wiki.python.org/moin/PythonBooks> | |