

## PYTHON FOR ENGINEERS

Course Code: CS1018

<b>Credits: 2</b>	<b>Teaching Scheme: Theory: 1 Hours / Week Laboratory: 2 Hours / Week</b>
<b>Section I</b>	
<p><b>Fundamentals of Python:</b> Features of Python, How to Run Python code, Identifiers, Reserved Keywords, Variables, Comments in Python, Indentation in Python, Input, Output statements.</p> <p><b>Operators:</b> Arithmetic Operators, relational Operators, Logical Operators, bitwise operators, other stray operators, Operator Precedence, Mathematical Functions, Trigonometric Functions, Random Number Functions, Strings in python: subscript operator, indexing, slicing a string, string methods</p> <p><b>Flow Control and Loops:</b> Decision Making: if statement, if...else statement, if..elif ..else statement, Nested if statement; Loops : while loop , for loop, range() function, continue and break statement, while and for loop with else statement, pass statement.</p>	
<b>Section II</b>	
<p><b>In-built Data Structures in Python:</b> List, Tuple, Set, Dictionary; Mutable and Immutable Objects, Data Type Conversion, Built-in methods and comprehensions.</p> <p><b>Functions:</b> Function Definition, Function Calling, Function Arguments, Anonymous Functions (Lambda Functions), recursive function, Function with more than one return value, default value to parameter, keyword parameter.</p> <p><b>File Handling:</b> Opening a File, Modes for Opening a File, Attributes of file object, closing a File, reading from a File, writing to a File, File Methods.</p> <p><b>NumPy and Matplotlib:</b> Introduction to NumPy, Arrays, Array indexing, comparison of Array and List data types, Array math, Broadcasting. Matplotlib -Plot, subplots and images.</p>	
<b>List of Practical:</b>	
<p><b>Assignments on Fundamentals of Python</b></p> <ol style="list-style-type: none"> <li>1. Program to Calculate the Area of a circle and triangle</li> <li>2. Program to Swap Two Variables</li> <li>3. Program to Generate a Random Number</li> <li>4. Program to Convert Kilometers to Miles</li> <li>5. Program to find maximum of two numbers</li> <li>6. Program to check if a number is even or odd</li> <li>7. Program to check if a number is positive, negative or 0</li> <li>8. A school decided to replace the desks in three classrooms. Each desk sits two students. Given the number of students in each class, print the smallest possible number of desks that can be purchased. The program should read three integers: the number of students in each of the three classes, a, b and c respectively. In the first test there are three groups. The first group has 20 students and thus needs 10 desks. The second group has 21 students, so they can get by with no fewer than 11 desks. 11 desks is also enough for the third group of 22 students. So we need 32 desks in total.</li> <li>9. H hours, M minutes and S seconds are passed since the midnight (<math>0 \leq H &lt; 12</math>, <math>0 \leq M &lt; 60</math>, <math>0 \leq S &lt; 60</math>). Determine the angle (in degrees) of the hour hand on the clock face right now.</li> <li>10. Given integer coordinates of three vertices of a rectangle whose sides are parallel to the coordinate axes, find the coordinates of the fourth vertex of the rectangle. In the first test the three given vertices are (1, 4), (1, 6), (7, 4). The fourth vertex is thus (7, 6).</li> <li>11. There was a set of cards with numbers from 1 to N. One of the card is now lost. Determine the</li> </ol>	

number on that lost card given the numbers for the remaining cards.

Given a number  $N$ , followed by  $N - 1$  integers - representing the numbers on the remaining cards (distinct integers in the range from 1 to  $N$ ). Find and print the number on the lost card.

**Assignments on Operators**

12. Program to find maximum of three numbers
13. Program to check if a year is leap
14. Program to check if a date is valid
15. Program to find the roots of a quadratic equation
16. Given a string. Delete from it all the characters whose indices are divisible by 3.
17. Given a sequence of integer numbers ending with the number 0. Determine the length of the widest fragment where all the elements are equal to each other.
18. In bowling, the player starts with 10 pins at the far end of a lane. The object is to knock all the pins down. For this exercise, the number of pins and balls will vary. Given the number of pins  $N$  and then the number of balls  $K$  to be rolled, followed by  $K$  pairs of numbers (one for each ball rolled), determine which pins remain standing after all the balls have been rolled. The balls are numbered from 1 to  $N$  (inclusive) for this situation. The subsequent number pairs, one for each  $K$  represent the start to stop (inclusive) positions of the pins that were knocked down with each role. Print a sequence of  $N$  characters, where "I" represents a pin left standing and "." represents a pin knocked down.
19. A timestamp is three numbers: a number of hours, minutes and seconds. Given two timestamps, calculate how many seconds is between them. The moment of the first timestamp occurred before the moment of the second timestamp.

**Assignments on Flow Control and Loops**

20. Program to display first  $n$  numbers
21. Program to calculate factorial of a numbers
22. Program to display numbers in the reverse order
23. Program to check if a number is prime
24. Program to calculate sum and average of first  $n$  numbers
25. Program to display first  $n$  multiples of a number
26. Program to display first  $n$  Fibonacci numbers
27. Program to find the sum of digits of a number

**Assignments on In-built Data Structures**

28. Program to Create and view elements of a list
29. Program to Create and view elements of a tuple
30. Program to access List Index and Values
31. Program to add two Lists
32. Program to check if a List is Empty or Not
33. Program to Find the Largest Number in a List
34. Program to Find the Second Largest Number in a List
35. Program to Put Even and Odd elements in a List into Two Different Lists
36. Program to Find all Numbers in a Range which are Perfect Squares and Sum of all Digits in the Number is Less than 10
37. Program to Generate Random Numbers from 1 to 20 and Append Them to the List
38. Program to Remove the Duplicate Items from a List
39. Program to create and view a dictionary
40. Program to create and view elements of a set

41. Program to Print values of dictionary
42. Program to print all keys of dictionary
43. Program to insert and delete from dictionary
44. Program to sort (ascending and descending) a dictionary by value
45. Program to concatenate dictionaries to create a new one
46. Program to check whether a given key already exists in a dictionary.
47. Program to merge two Python dictionaries
48. Program to get the maximum and minimum value in a dictionary
49. Program to Add a list of elements to a set
50. Program to Update the first set with items that don't exist in the second set
51. Program to Return a set of elements present in Set A or B, but not both
52. Program to check if two sets have any elements in common.
53. Program to Remove items from set1 that are not common to both set1 and set2

#### Assignments on Functions

54. Function to check if a number is even or odd
55. Function to find the maximum of two numbers
56. Function with keyword arguments
57. Function with default arguments
58. Function to Find the Factorial of a Number Using Recursion
59. Function to Find the Sum of the Digits of the Number Recursively
60. Create a function named count vowels that accepts a string and returns the number of vowels in the string.
61. Write a function greet user that takes a user's name as a parameter and prints a personalized greeting message.
62. Given a sequence of integers that end with a 00. Print the sequence in reverse order. Don't use lists or other data structures. Use the force of recursion instead.

#### Assignments on File Handling

63. To keep record of students' data, manipulate files to store, update, and delete students' information.
64. Write a Python program that reads the content of a text file and prints it to the console.
65. Create a Python program that takes user input (a string) and writes it to a new text file.
66. Write a Python program that reads a text file, counts the number of words in it, and writes the word count to a new file.
67. Write a Python program that reads the content of one text file and writes it to another file.
68. Develop a Python program that counts the number of lines in a text file and displays the count.
69. Write a Python program that appends a user-provided string to the end of an existing text file.
70. To keep record of patients' medical data, manipulate files to store, update, and delete such information.
71. Write a python program to create two 3X3 random matrixes and perform following operation: (a) Addition (b) subtraction (c) multiplication and display shape, dimensions, dtype, Rank and flatten output of every o/p matrix.
72. Write a Python program to plot line chart, bar chart, pi chart, scatter chart, histogram for taking two different arrays as input.

**Link of YouTube lectures: Python for beginners**

[www.youtube.com/watch?v=QXeEoD0pB3E&list=PLsyebzWxl7poL9JTVyndKe62ieoN-MZ3](https://www.youtube.com/watch?v=QXeEoD0pB3E&list=PLsyebzWxl7poL9JTVyndKe62ieoN-MZ3)

**List of Projects Areas:**

Students will be doing course projects in different areas of application of python programming such as web application, data analysis, computer vision, machine learning, artificial intelligence, web technologies, numerical analysis etc.

**Text Books:**

1. "Programming And Problem Solving With Python", Ashok Kamthane and Amit Ashok Kamthane , ISBN 9789387067578
2. "Beginning Programming with Python for Dummies", John Paul Mueller , Wiley Publishing, ISBN 9788126553488

**Reference Books:**

1. Dive into Python 3: Mark Pilgrim, Jesse Noller, Wiley Publishing , ISBN 9788184899115.
2. Python in Easy Steps, Tata McGraw Hill Education, ISBN 9789351343080
3. Learning Python, 5th Edition, Mark Lutz, O'Reilly, ISBN 978-1449355739

**Moocs Links and additional reading material:** [www.nptelvideos.in](http://www.nptelvideos.in), <https://snakify.org/en/>

**Course Outcomes:**

The student will be able to -

2. understand Python programming basics.
3. demonstrate mathematical and string manipulation functions in Python.
4. apply core Python scripting elements such as flow control structures and loops.
5. develop essential skills in python programming concepts like data structures and different built-in functions.
6. apply a modular programming approach by making use of functions.
7. demonstrate the ability to data frames, plots and files in different modes.