## Guidelines

| Semester | DSC No. | Title | L | T* | P* | Total credits | Pre-requisites |
|---|---|---|---|---|---|---|---|
| **I** | DSC-1 | Programming using Python | **3** | **0** | **1** | **4** | None |

| S. No. | Unit Name | Chapters | References | Weeks |
|---|---|---|---|---|
| 1. | Unit 1 Introduction to Programming | 2<br>1 | [2]<br>[1] | 1-2 |
| 2. | Unit 2 Creating Python Programs | 2 (2.1 to 2.3),<br>3 | [1] | 3-6 |
| 3. | Unit 3 Built-in data structures | 6 (6.1, 6.2),<br>7 (except 7.1.11) | | 7-11 |
| 4. | Unit 4 Object Oriented Programming | 10 (10.1, 10.2) | | 12-13 |
| 5. | Unit 5 File and exception handling | 9 (except 9.2 and 9.5) | | 14-15 |

References


1. Taneja, S., Kumar, N., *Python Programming- A modular Approach*, Pearson Education India, 2018.
2. Balagurusamy E., *Introduction to Computing and Problem Solving using Python*, 2nd edition, McGraw Hill Education, 2018.


Additional References


1. Brown, Martin C., *Python: The Complete Reference*, 2nd edition, McGraw Hill Education, 2018.
2. Guttag, J.V. *Introduction to computation and programming using Python*, 2nd edition, MIT Press, 2016.


**Practical List**

1. WAP to find the roots of a quadratic equation

2. WAP to accept a number ‘n’ and

   a. Check if ’n’ is prime

   b. Generate all prime numbers till ‘n’

   c. Generate first ‘n’ prime numbers

   This program may be done using functions

3. WAP to create a pyramid of the character ‘*’ and a reverse pyramid

```
    *
   ***
  *****
 *******
*********

*********
 *******
  *****
   ***
    *
```

4. WAP that accepts a character and performs the following:

   a. print whether the character is a letter or numeric digit or a special character

   b. if the character is a letter, print whether the letter is uppercase or lowercase

   c. if the character is a numeric digit, prints its name in text (e.g., if input is 9, output is NINE)

5. WAP to perform the following operations on a string

   a. Find the frequency of a character in a string.

   b. Replace a character by another character in a string.

   c. Remove the first occurrence of a character from a string.

   d. Remove all occurrences of a character from a string.

6. WAP to swap the first n characters of two strings.

7. Write a function that accepts two strings and returns the indices of all the occurrences of the second string in the first string as a list. If the second string is not present in the first string then it should return -1.

8. WAP to create a list of the cubes of only the even integers appearing in the input list (may have elements of other types also) using the following:

   a. 'for' loop

b. list comprehension

9. WAP to read a file and
    a. Print the total number of characters, words and lines in the file.
    b. Calculate the frequency of each character in the file. Use a variable of dictionary type to maintain the count.
    c. Print the words in reverse order.
    d. Copy even lines of the file to a file named ‘File1’ and odd lines to another file named ‘File2’.
10. WAP to define a class Point with coordinates x and y as attributes. Create relevant methods and print the objects. Also define a method distance to calculate the distance between any two point objects.
11. Write a function that prints a dictionary where the keys are numbers between 1 and 5 and the values are cubes of the keys.
12. Consider a tuple t1=(1, 2, 5, 7, 9, 2, 4, 6, 8, 10). WAP to perform following operations:
    a. Print half the values of the tuple in one line and the other half in the next line.
    b. Print another tuple whose values are even numbers in the given tuple.
    c. Concatenate a tuple t2=(11,13,15) with t1.
    d. Return maximum and minimum value from this tuple
13. WAP to accept a name from a user. Raise and handle appropriate exception(s) if the text entered by the user contains digits and/or special characters.