

1

22

333

4444

55555

The number of rows n , is to be taken from the user.

3. Write a program to compute the factors of a given number.
4. Write a menu driven program to perform the following operations on an array:
 - a. Find the minimum, maximum and average of the array elements
 - b. Search an element in the array using linear and binary search
5. Write a menu driven program to perform the following operations on a string:
 - a. Calculate length of the string
 - b. Check whether the first character of every word in the string is in uppercase or not
 - c. Reverse the string
6. Create a class Triangle. Include overloaded functions for calculating the area of a triangle.
7. Create a template class TwoDim which contains x and y coordinates. Define default constructor, parameterized constructor and void print() function to print the co-ordinates. Now reuse this class in ThreeDim adding a new dimension as z . Define the constructors and void print() in the subclass. Implement main() to show runtime polymorphism.

GE1b: Programming with Python

Course Objective

The course is designed to introduce programming concepts using Python to students. The course aims to develop structured as well as object-oriented programming skills using Python. The course also aims to achieve competence amongst its students to develop correct and efficient Python programs to solve problems in their respective domains.

Course Learning Outcomes

On successful completion of the course, students will be able to:

1. Write simple programs using built-in data structures in Python.
2. Implement arrays and user defined functions in Python.
3. Solve problems in the respective domain using suitable programming constructs in Python.
4. Solve problems in the respective domain using the concepts of object oriented programming in Python.

Syllabus

Unit 1 Introduction to Programming: Problem solving strategies; Structure of a Python program; Syntax and semantics; Executing simple programs in Python.

Unit 2 Creating Python Programs: Identifiers and keywords; Literals, numbers, and strings; Operators; Expressions; Input/output statements; Defining functions; Control structures (conditional statements, loop control statements, break, continue and pass, exit function), default arguments.

Unit 3 Built-in data structures: Mutable and immutable objects; Strings, built-in functions for string, string traversal, string operators and operations; Lists creation, traversal, slicing and splitting operations, passing list to a function; Tuples, sets, dictionaries and their operations.

Unit 4 File and exception handling: File handling through libraries; Errors and exception handling.

References

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

Additional References

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

Suggested 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
 - d. 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. Write a function that prints a dictionary where the keys are numbers between 1 and 5 and the values are cubes of the keys.
11. 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 $t_2=(11,13,15)$ with t_1 .
 - d. Return maximum and minimum value from this tuple
12. 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.

GE2a: Data Analysis and Visualization using Python

Course Objective

This course is designed to introduce the students to real-world data analysis problems, their analysis and interpretation of results in the field of exploratory data science using Python.

Course Learning Outcomes

On successful completion of the course, students will be able to:

1. Apply descriptive statistics to obtain a deterministic view of data
2. Perform data handling using Numpy arrays
3. Load, clean, transform, merge and reshape data using Pandas
4. Visualize data using Pandas and matplotlib libraries

Syllabus

Unit 1 Introduction to basic statistics and analysis: Fundamentals of Data Analysis, Statistical foundations for Data Analysis, Types of data, Descriptive Statistics, Correlation and covariance, Linear Regression, Statistical Hypothesis Generation and Testing, Python Libraries: NumPy, Pandas, Matplotlib

Unit 2 Array manipulation using Numpy: Numpy array: Creating numpy arrays, various data types of numpy arrays, indexing and slicing, swapping axes, transposing arrays, data processing using Numpy arrays

Unit 3 Data Manipulation using Pandas: Data Structures in Pandas: Series, DataFrame, Index objects, Loading data into Pandas data frame, Working with DataFrames: Arithmetics,