



## End Term (Odd) Semester Examination Dec 2025

Roll no.....

Name of the Program and semester: MCA (AIDS)

Name of the Course: Python Programming

Course Code: TMD 102

Time: 3-hour

Maximum Marks: 100

### Note:

- (i) All the questions are compulsory.
- (ii) Answer any two sub questions from a, b and c in each main question.
- (iii) Total marks for each question are 20 (twenty).
- (iv) Each sub-question carries 10 marks.

Q1.

(2X10=20 Marks)

a. Explain the different conditional statements in Python. Explain the usage of range() function. Write a python program to find all the palindrome numbers up to n where n is taken from user. Co1

b Define short circuiting techniques in python? Explain it by taking the example of all the logical operators. Co1

c Explain about how exceptions are handled in python? Define the various blocks briefly. Write a python program to handle ZeroDivision Error. Co1.

Q2.

(2X10=20 Marks)

a. What are Lambda functions? Explain how lambda functions differ from user defined functions. Define a lambda function to print the square of all the numbers in a list. Co2

b. Write a Python program that accepts a multi-line text input from the user and performs the following operations:

1. Removes all punctuation marks and converts the text to lowercase.
2. Identifies all unique words and counts their frequencies.
3. Displays the top five most frequent words along with their counts.

Co2

c. i. Explain the concept of **mutability** in Python with reference to lists. How does it differ from the behavior of tuples? Give suitable examples.

ii. Write a Python program that accepts a list of integers and performs the following operations:

1. Removes duplicate elements without using a set.
2. Sorts the unique elements in descending order using a user-defined function.
3. Calculates and displays the sum and average of the final list.

Co2

Q3.

(2X10=20 Marks)

a. Explain the concept of **key-value mapping** in Python dictionaries. How does elements are accessed in dictionary? Write a Python program that accepts a paragraph as input and creates a dictionary where:

1. Each **unique word** is a key.
2. The **frequency of the word** is the corresponding value.



### End Term (Odd) Semester Examination Dec 2025

3. The program then displays the top three most frequent words with their counts. Co3
- b. Explain the different modes of opening a file. Write a python program to copy the content of one file to another. Co3
- c. Explain the key properties of **sets** in Python. How do they differ from **lists** and **tuples** in terms of mutability, ordering, and duplication? Illustrate with suitable examples. Explain any three methods used for sets operation. Co3

Q4. (2X10=20 Marks)

- a. Discuss how instance variables and class variables differ in terms of scope and memory allocation in Python classes. Write a Python program to define a class **Rectangle** with: Co4
1. A **constructor** that initializes the **length** and **breadth** of the rectangle.
  2. A method **area()** that returns the area of the rectangle.
  3. A method **perimeter()** that returns the perimeter of the rectangle.
  4. A suitable main section to create two rectangle objects and display their area and perimeter.
- b. Write a python program to overload power(\*\*) operator to find the power of one object to the power of other object. Co4
- c. Explain the concept of **inheritance** in Python. How does it promote code reusability? Write a Python program to demonstrate **constructor invocation in inheritance**: Co4
- Create a base class **Person** with a constructor that initializes name and age, and a method **display()** to show these details.
  - Derive a subclass **Student** that inherits from **Person** and has an additional attribute **student\_id**.
  - In the **Student** class constructor, invoke the **base class constructor** using **super()** and display all details of the student using an overridden **display()** method.

Q5. (2X10=20 Marks)

- a. Explain the architecture of the Tkinter module in Python. How does the **mainloop()** function control the GUI event handling process? Compare and contrast the **pack()**, **grid()**, and **place()** geometry managers in Tkinter with examples of suitable use cases. Co5
- b. Write a Python program using Tkinter to create a simple GUI application that contains:
- A Label displaying "Enter your name:"
  - An Entry widget for user input
  - A Button labeled "Greet" that, when clicked, displays a greeting message (e.g., *Hello, Anushka!*) using a Label below the button. Co5
- c. Explain the concept of **event-driven programming** in Tkinter. How are events like button clicks or key presses handled? Write a Python program using Tkinter to design a simple **Temperature Converter** application that:
- Takes input in Celsius from the user.
  - Converts it to Fahrenheit on button click.
  - Displays the result in a Label widget Co5