Roll No. | | | | | | | | | | | | **Total No. of Pages : 02**

Total No. of Questions : 07

BCA/B.Sc. (IT) (Sem.–3)

PROGRAMMING IN PYTHON

Subject Code : UGCA-1914

M.Code : 78180

Date of Examination : 20-06-2025

Time : 3 Hrs. **Max. Marks : 60**

INSTRUCTIONS TO CANDIDATES :

1. **SECTION-A is COMPULSORY consisting of TEN questions carrying TWO marks each.**
2. **SECTION-B contains SIX questions carrying TEN marks each and students have to attempt any FOUR questions.**

SECTION-A

1. Write briefly :

a) Name three limitations of Python.

b) How do you set up environment variables for Python?

c) What is exception handling in Python?

d) Give an example of a real-world application of Python.

e) Name three native data types in Python.

f) What does the range() function do in Python?

g) What is an anonymous function in Python?

h) How do you define a user-defined function in Python?

i) What is the purpose of directories in Python?

j) What is file encoding and why is it important?

SECTION-B

2. Provide an in-depth analysis of Python's applications across various domains, highlighting specific examples and discussing its advantages over other programming languages in these contexts.

3. Explain the concept of control structures in Python, discussing how they enable the flow of program execution. Provide examples of conditional statements, loops and function calls to illustrate their usage.

4. Investigate the features and usage of dictionaries in Python, emphasizing their key-value pair structure and efficient data retrieval capabilities. Provide examples to illustrate dictionary operations, including adding/removing items, accessing values and iterating through key-value pairs.

5. Compare and contrast pass by value and pass by reference in Python, explaining how arguments are passed to functions and how changes made to mutable objects within functions can affect the original objects. Provide examples to demonstrate the differences between pass by value and pass by reference.

6. Explain the process of working with files in Python, including file, opening, reading, writing, and closing operations. Describe various file opening and discuss their respective purposes and behaviors. Additionally, discuss file attributes such as mode, name, closed, and encoding and demonstrate how to access them using file object attributes.

7. Explore garbage collection in Python, explaining how it manages memory by automatically reclaiming unused objects to free up system resources. Discuss the garbage collection mechanisms used in Python, including reference counting and cyclic garbage collection, and explain how they work to identify and remove unreachable objects from memory. Additionally, discuss techniques for controlling garbage collection behavior and optimizing memory usage in Python programs.