Prof. A N Ramya Shree
CSE, RNSIT

ADP-15CS55-Question Bank

Module-4

- Define class and object. Given an example for creating a class and an object of that class.
- What are attributes? Explain with an example and respective object diagram.
- Explain class attributes and instance attributes with examples.
- Write a program to create a class called Point with two attributes x and y. Write following functions and demonstrate the working of these functions by creating suitable objects.
    - To read attribute values
    - To display point as an ordered pair
    - To find distance between two points
- Write a program to create a class called Rectangle with the help of a corner point, width and height. Write following functions and demonstrate their working:
    - To find and display center of rectangle
    - To display point as an ordered pair
    - To resize the rectangle
- Differentiate copy.copy() and copy.deepcopy() with suitable examples.
- Discuss the methods isinstance(), hasattr() and vars() with suitable examples.
- Differentiate pure functions and modifiers with suitable examples
- List out the object oriented characteristics possessed by Python.
- Differentiate methods and functions.
- Discuss the significance of __init__() method & __del__() in Python with a proper example code snippet.
- Briefly discuss self keyword with example in Python.
- What does __str__() method do in Python? Illustrate with a code snippet.
- Discuss operator overloading. Mention any five operators with respective special functions to be overloaded in Python.
- Write a program to add two point objects by overloading + operator. Overload __str__() to display point as an ordered pair.
- Write a program to create a class Time to represent time in HH:MM:SS format. Perform following operations:
    - Overload + to add two time objects
    - Overload + to add a numeric value to a time object (commutative)
    - Overload __str__() to display time in appropriate format
- What do you mean by “instance as returning value”? Explain with an example.
- Justify the statement “Objects are mutable”.
- What is an embedded object? Give an example.
- When do we encounter AttributeError?
- How do you find the memory address of an instance of a class?
- Define Inheritance.
- Explain different types of types of inheritance with examples.
- What is the significance of type based dispatch in Python? Explain with example.
- Describe Encapsulation in Python with example.

- What are access modifiers? Explain with example.
- What is operator overloading? Explain Python built in Operator overloading methods with examples.
- Explain Stack, Class, Object and State Diagram with examples.
- Describe the significance of super() function with examples
- Explain Method Overriding with an example
- List and explain different access specifiers in Python with examples
- Explain Multiple Inheritance with an example
- What is MRO?,Mention its significance in Multiple Inheritance
- What is Multilevel Inheritance and explain use of super () in it.
- Explain with an example Passing object as parameter
- Explain with an example returning the object from the class method
- Explain with an example instances as return values
- Explain with an example accessing the objects by outside methods
- Describe encapsulation in Python with example.
- Programs Related to Inheritance, Polymorphism – overloading & overriding, Encapsulation OOP Concepts.
- A rectangle can be created using any of the following data –By knowing width and height of a rectangle and one corner point (ideally, a bottomleft corner) in a coordinate system , By knowing two opposite corner points. Write a class Rectangle containing numeric attributes width and height. This class should contain another attribute corner which is an instance of another class Point.
    - Increase the size of rectangle
    - Move the rectangle by certain distance
    - Check whether point coordinates are equal
    - Midpoint of two points
- Write a class Point representing a point on coordinate system. Implement following functions –
    - A function read_point() to receive x and y attributes of a Point object as user input.
    - A function distance() which takes two objects of Point class as arguments and computes the Euclidean distance between them.
    - A function print_point() to display one point in the form of ordered-pair.