Part A: Conceptual Question

1. Definition of a Class and an Object
   - What is a class in object-oriented programming?
     A class is a user-defined data type that we can use in our program, and it works as an object constructor, or a "blueprint" for creating objects.
   - What is an object, and how does it relate to a class?
     An object contain information like attributes, smethods,.In C++, an object is created from a class.
2. Constructors and Destructors
   - Define a constructor. What is its role in a class?
     A constructor in C++ is a special method that is automatically called when an object of a class is created.
   - Define a destructor. Why is it important in managing an object's lifecycle?
     In C++, a destructor is a specific member function that is automatically called when an object is intentionally destroyed or exits scope. It is employed to relinquish resources, such memory or file handles, that the object may have accumulated over time.
3. Object Lifecycle
   - Briefly describe the lifecycle of an object from instantiation to destruction.
     The object
     First the object is create in memory, then we call a constructor to initializes the object with default or input values. Once the constructor runs, the object will allocate resources like memory and files. The object is used by calling its methods, accessing its properties, or interacting with it in other ways. It continues to exist as long as there are references to it. When the object is no longer needed, the destructor or finalizer is invoked. This step releases the resources used by the object, like memory, file, or network connections.
   - Why is it important for a class to manage its resources (e.g., memory) during its lifecycle?
     Effective resource management ensures that the program runs efficiently, safely, and without resource-related issues.