A black background with a black square

Description automatically generated with medium confidence

**School of Computing, Engineering and Physical Sciences**

**MSc Information Technology**

**COMP11124 Information Technology Project Management**

**Class Test Student Written Answer Sheet**

**Session 2023/2024 Term 2**

Academic Integrity Statement:

‘I certify that all material in this submission is my own work. I have read and understood the

section in the university regulations regarding collusion, cheating and plagiarism. I confirm

I have not purchased/commissioned material which is presented as my own, including the

use of online services’

Banner ID: B01730479

Name: Sabarish Jayesh Kumar

|  |
| --- |
| (7) Evaluate critically why following the Object-Oriented Paradigm for this library application is more suitable over a procedural approach. (8 Marks) |
| We use OOP for the following functionality as these are the main advantages and are unique to OOP Paradigm.   1. **Inheritance**: by using different base classes and sub classes to inherit attributes and functions. We used the base class Rentable object as a parent class and shared its attributes and functions to its base classes Book and DVD 2. **Encapsulation**: we used the **RentableObject** class to encapsulate attributes like name, available, and rental\_history, along with methods like availability, show, rent\_item, and return\_item. 3. **Polymorphism:** We used the Book and DVD subclasses override the show method to provide specialized behavior while maintaining a consistent interface. We also used a for loop to itereate through the show function defining it in all the classes 4. **Abstraction:** The **RentableObject** class defines an abstract method **rent\_item,** which must be implemented by subclasses. This makes sures some details are hidden but makes sure other classes adhere to the principles of rent\_iem. 5. **Code Reusability & Modularity**: we can use the classes in a different file calling is classes.py and the main functions is main.py and use the classes.py file in a modular model and reuse it |
| (8) If you were to improve this small program, what extra steps would you take (think about SOLID principles or any other OOP topics covered)? (5 Marks) |
| If I had to improve for proper deployment and wanted to take this is as a large scale program, I would have done the following using SOLID principles   1. **Single Responsibility Principle (SRP):** we can limit the **RentableObject** class to manage rental objects exclusively, while other tasks like displaying items or managing rentals can be delegated to different other classes. 2. **Open/Closed Principle (OCP):** Instead of directly altering methods like show in subclasses such as Book and DVD, we can explore other functionalities of the show method separately. 3. **Liskov Substitution Principle (LSP):** we can make sure that the subclass can replace the base class easily.so any function of Parent class, i.e, Rental\_Object can be used in any of the child classes. 4. **Interface Segregation Principle (ISP):** we can have different interfaces, like one for DVD, one for Books to cater to different audience |