UNIVERSITY OF CALOOCAN CITY
COMPUTER ENGINEERING DEPARTMENT

Data Structure and Algorithm

Laboratory Activity No. 1

Object-oriented Programming

Submitted by:
Uy, Junichiro H.

Instructor:
Engr. Maria Rizette H. Sayo

July, 27, 2025

I. Objectives

This laboratory activity aims to implement the principles and techniques in object-oriented programming specifically through:

- Identifying object-orientation design goals
- Identifying the relevance of design pattern to software development

II. Methods

- Software Development
    - The design steps in object-oriented programming
    - Coding style and implementation using Python
    - Testing and Debugging
    - Reinforcement of below exercises

A.   Suppose you are on the design team for a new e-book reader. What are the primary classes and methods that the Python software for your reader will need? You should include an inheritance diagram for this code, but you do not need to write any actual code. Your software architecture should at least include ways for customers to buy new books, view their list of purchased books, and read their purchased books.

B.   Write a Python class, Polygons that has three instance variables of type str, int, and float, that respectively represent the name of the polygon, its number of sides, and its area. Your class must include a constructor method that initializes each variable to an appropriate value, and your class should include methods for setting the value of each type and retrieving the value of each type.
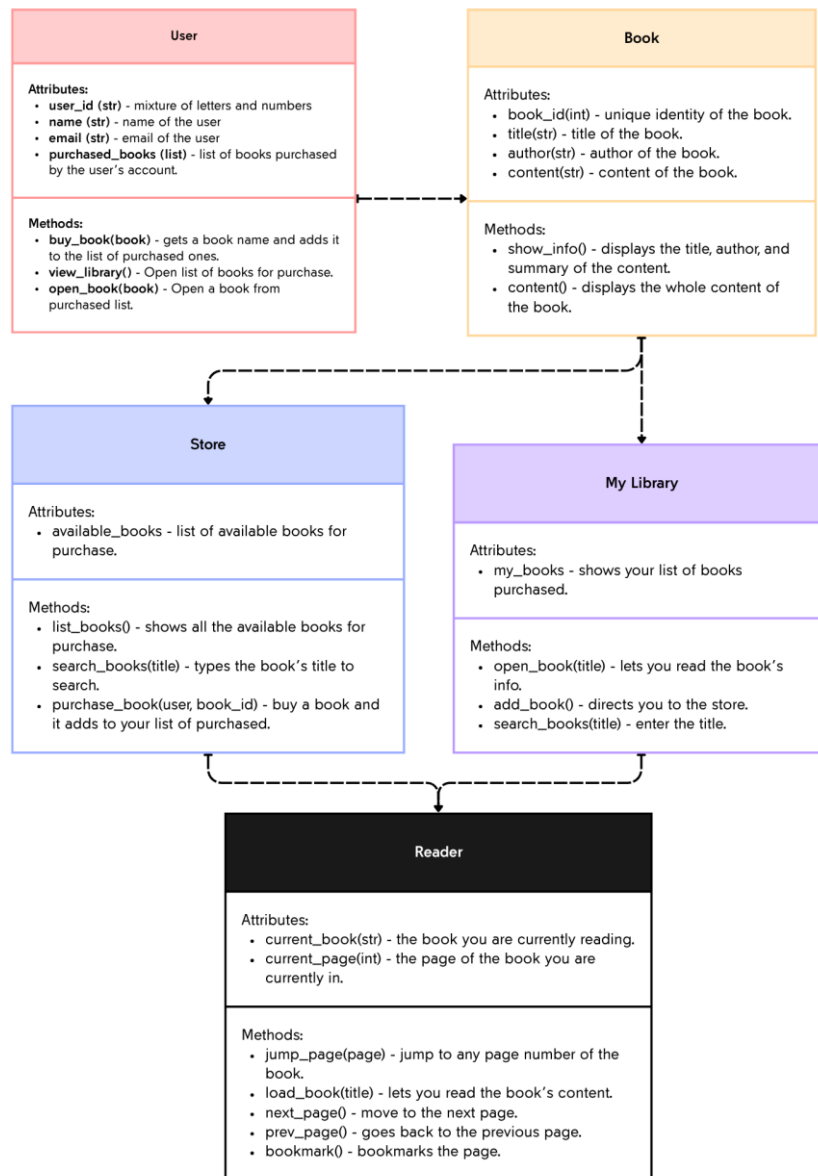
III. Results

A.

Figure 1: Screenshot of diagram made in canva

This program is about how an e-book reader with a built-in shop would work. This program gives you the power to read e-books with ease, get e-books online, and explore other themes with just a few clicks. The user interface allows you to see your account id, nickname, and email so you can get notifications and shows you a button that can open your library, which shows the books you purchased. Then comes the book, which contains information regarding its author, title, its unique identification, and a short summary that gives a quick overview of the topic the book contains. Next would be the store that can be accessed through clicking the shop button, the shop interface displays lists of books, which we can apply a sorting algorithm to make it organized by alphabetical order, price, or recency. Next would be your library, where

the book will go in once you purchase it. Both this interfaces contains search inputs that allows you to find the book your looking for with ease. Connecting these two would be the reader class, which allows you to read its contents, it also has next and previous buttons and a jump to a page button that allows you to change pages, and it also has a bookmark page function that allows you to bookmark a page that you want to go back in, or I can add an automated bookmark function that when the book is closed and you open it again, the page you were in before would be the one to reload.

B.

```python
class Polygons:
  def __init__(self, name: str, sides: int, area: float):
    self._name = name
    self._sides = sides
    self._area = area

  #allows you to assign new parameters
  def new_name(self, name: str):
    self._name = name

  def new_sides(self, sides: int):
    self._sides = sides

  def new_area(self, area: float):
    self._area = area

  #allows you to call them to display
  def show_name(self):
    return self._name

  def show_side(self):
    return self._sides

  def show_area(self):
    return self._area
```

Figure 2: Screenshot of Source Code from Google Colab

This program allows you to set a name, number of sides, and area. It takes a string (str), Integer (int), and Float as parameters and we can call its function whenever we want. As you can see from the image, I was able to give it functions using the new and show variables. I call the show variables when I need to print it and use the new_variables to change its property.

```
Square = Polygons("Mr Boxy", 4, 25.5)
Triangle = Polygons("Spikey Mike", 3, 16.6)

print(f"""Hello, I am a Square and you can call me {Square.show_name()},
I have {Square.show_side()} sides, and my area would be around
{Square.show_area()}! Yeah I know I'm overweight...""")
print(" ")
print(f"""Hi, I'm also a Polygon which has {Triangle.show_side()},
also know as a Triangle! My name would be {Triangle.show_name()}
and I cover about {Triangle.show_area()}.""")
print(" ")

print(f"{Square.show_name()} has started exercising a while ago...")
Square.new_area(10.5)

print(f"""After all that running, I managed to lose 15cm of fat around my plane,
now I only measure to {Square.show_area()}!""")
```

```
Hello, I am a Square and you can call me Mr Boxy,
I have 4 sides, and my area would be around
25.5! Yeah I know I'm overweight...

Hi, I'm also a Polygon which has 3,
also know as a Triangle! My name would be Spikey Mike
and I cover about 16.6.

Mr Boxy has started exercising a while ago...
After all that running, I managed to lose 15cm of fat around my plane,
now I only measure to 10.5!
```

Figure 3: Screenshot of Source Code with output from Google Colab

From the code I showed, I used the values I assigned to each parameter. In a sense, I was able to use the same commands and assign different values to each variable thanks to the inheritance property of classes. With that in mind, I was able to form a sentence using the values I assigned.

IV. Conclusion

In conclusion, I was able to learn how to use inheritance for my own advantage. I also learned how to give structure to an application and what its attributes and methods are that I am going to use, and I can see it helping me more in the future when I start making my own applications. This way of programming could potentially make my work faster and more efficient, as I would have a clearer outline of what I need to get done than if I just started somewhere, praying that I would be able to finish the whole application in just one go, starting from a small part and scaling it. This is helpful for a person with a human brain.

References

[1] Co Arthur O.. "University of Caloocan City Computer Engineering Department Honor Code," UCC-CpE Departmental Policies, 2020.