Data Structure and Algorithm

Laboratory Activity No. 1

Object-oriented Programming

|  |  |
| --- | --- |
| *Submitted by:* | *Instructor:* |
| t, FirstName MI. | Engr. Maria Rizette H. Sayo |

July, 26, 2005

# 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

# Methods

* Software Development
  + The design steps in object-oriented programming
  + Coding style and implementation using Python
  + Testing and Debugging
  + Reinforcement of below exercises
  1. 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.
  2. 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.

# Results

A diagram of a computer

AI-generated content may be incorrect.

The diagram shows how a book system works. A **User** has a name and an email. The user can buy books, see the books they have purchased, and read them.

The **Book Store** has a list of books available to buy. Books can be added to the store, users can look at all the books, and they can purchase any book they like.

A **Book** has basic information such as a title, author, and price. You can view the details of a book.

An **Audio Book** is a type of book that you can listen to. It also includes the length of the audio in hours.

An **E-Book** is a digital version of a book. It includes the format of the file, like PDF or EPUB.

An **E-Book Reader** is a device used to read or listen to e-books and audio books. It has a name, screen size, and battery life. The reader can open books and show their content on the screen.

In the diagram, each box has **attributes**, which are the details or features it has, and **methods**, which are the actions it can do.

A screenshot of a computer program

AI-generated content may be incorrect.

A white background with black text

AI-generated content may be incorrect.

The Polygons class is used to create objects that represent polygons. Each polygon has three attributes: a name (like "Triangle"), the number of sides (an integer), and the area (a decimal number). The class has a constructor method that sets these attributes when a new polygon is created. It also has setter methods to change the values of the name, sides, and area, and getter methods to retrieve these values. In the test code, a polygon is created as a triangle with 3 sides and an area of 15.5. The program prints these details, then updates the polygon to a square with 4 sides and an area of 20.0, and prints the updated details. When the number of sides is changed, the program also prints a message showing the new value. This class helps organize and manage information about polygons in a clear and simple way.

Figure 1 Screenshot of program

If an image is taken from another literature or intellectual property, please cite them accordingly in the caption. Always keep in mind the Honor Code [1] of our course to prevent failure due to academic dishonesty.

# Conclusion

The conclusion expresses the summary of the whole laboratory report as perceived by the authors of the report.

The Polygons class and its test code show how to use Python classes to organize and manage data about shapes. By defining attributes and using methods to set and get values, the class makes it easy to create polygon objects and change their details when needed. This approach helps keep the code clear and makes working with many polygons simple and organized. Overall, it demonstrates basic but important concepts of object-oriented programming in a practical way.

**References**

[What is Polygenic Inheritance? - GeeksforGeeks](https://www.geeksforgeeks.org/biology/polygenic-inheritance/)