Multi-Paradigm Programming Shop Assignment Report

Name: Megan O'Donovan
Student ID: G00411435

Introduction

The aim of this project was to build on the functionality of the programming used in the lectures on Shop data. The report compares the implementation of two paradigms, Objected Orientated Programming using Python and Procedural Programming using Python and C Programming.

Functionality

This project looks at simulating a shop experience and processing customers' orders. The report is an extension of programming used in the lectures for Multi Paradigm Programming. The shop contains a list of available items and initial cash. The program allows the end users to purchase items using two methods: by reading the customers data from a csv file or by live interaction where the user is prompted to input information about their order. Any successful purchase will update the shops stock and cash flow while any unsuccessful purchase will generate an error message relating to insufficient funds or stock. Each program's functionality should algin for each implementation.

Programming Paradigm

Programming Paradigms are different ways a script or programming language can be written. A means to classify programming languages according to their features. The imperative programming paradigm instructs the machine how to change its state. It can be procedural, where it groups instructions into procedures, or object oriented, where it groups instructions with the part of the state they operate on. Procedural and Object Oriented are known as Imperative Programming Paradigms. Procedural programming is where the problem is broken down into smaller sub-procedures. This process is called functional decomposition and is continued until the sub-procedures are solved. Object Oriented programming is where all computations are conducted using objects. In this report, languages C and Python are used to demonstrate procedural programming while Python is used to demonstrate object-oriented programming.

Procedural Programming (PP)

PP takes a top-down approach because the computer conducts the instructions in a well-defined sequence from beginning to end. The instructions are divided into smaller blocks of code known as functions where the program calls the right function at the right time. An advantage of procedural

programming is that it can follow a simple structure and functions can be reused within the program repeatedly.

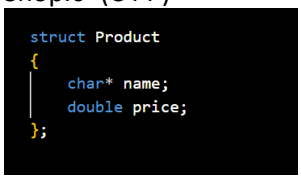
Object Orientated Programming (OOP)

OOP operates using a bottom-up approach, based around data or objects rather than functions and logic. An object can be defined as a data field that has a unique attribute and behaviour. Objects are an instance of a class; a class can have many objects. This approach to programming is well-suited for programs that are large, complex and actively updated or maintained.

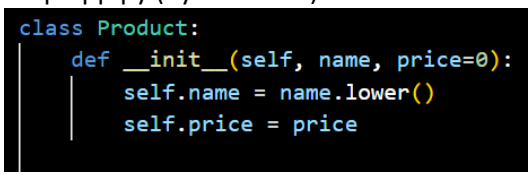
Similarities

- In shop.opp.py (OOP) uses classes and in the PP approach dataclasses (shop.py) are used while structs (shop.c) act as the class equivalent. Across the three programming the grouping of variables follows similar approaches.

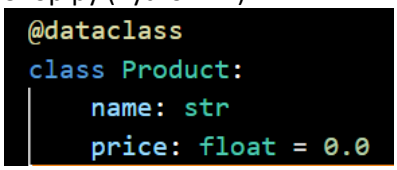
Shop.c (C PP)



Shop.opp.py (Python OOP)

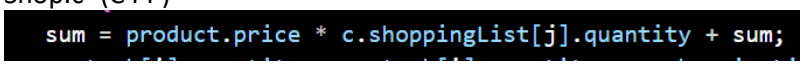


Shop.py (Python PP)

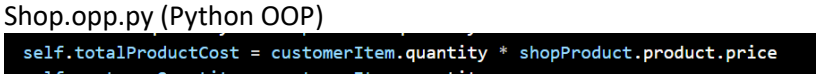


- Both paradigm programs split the code into smaller subqueries. This is achieved in OOP through using objects and in PP this is accomplished with procedures.

Shop.c (C PP)



Shop.opp.py (Python OOP)



Shop.py (Python PP)

- All three programs use the display menu to call methods and procedures, allowing the end user to select what shop scenario to execute.

Shop.c (C PP)

```c
if (choice == '1') { // Display shop //

    printShop(shop); // Print shop
    displayMenu(shop); // Return to main menu
}
```

Shop.opp.py (Python OOP)

```python
if (self.choice == "1"): # Display shop #

    print(self) # Print the shop
    self.displayMenu() # Return to main menu
```

Shop.py (Python PP)

```python
if choice == '1': # Display shop #

    printShop(shop) # Print the shop
    displayMenu(shop) # Return to main menu
```

- When executing the live shop scenario all three programs accept non case sensitive responses from the end user.

Shop.c (C PP)

```c
if (strcmp(product.name, productcustomer.name) == 0)
```

Shop.opp.py (Python OOP)

```python
self.name = name.lower()
```

Shop.py (Python PP)

```python
if item.product.name.lower() == customerItem.product.name.lower():
```

- OPP, PP and C programs will generate error messages if an end users' responses do not align with stock availability and customer funds. All versions use If statements to generate these error messages.

Shop.c (C PP)

```c
if(c.budget< sum)
{
    shortamount = (sum - c.budget);
    printf("-----------------------------------------------
    printf("We are sorry, but you do not have enough money
```

Shop.opp.py (Python OOP)

```python
if c.budget < self.totalProductCost:
    print("\n**********************************************
    print(f"\nWe are sorry, but you do not have enough money
    print("\n**********************************************
```

Shop.py (Python PP)

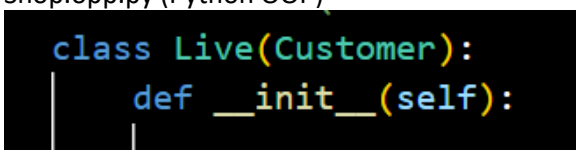
```python
    if customer.budget > orderTotal:
        # Update the shop stock
```

Differences

- As previously discussed, PP employs a top-down approach in which each instruction is carried out in order, in a systematic way so that a computer can understand what to do while OOP takes a bottom-up approach where the data takes precedence of the order.

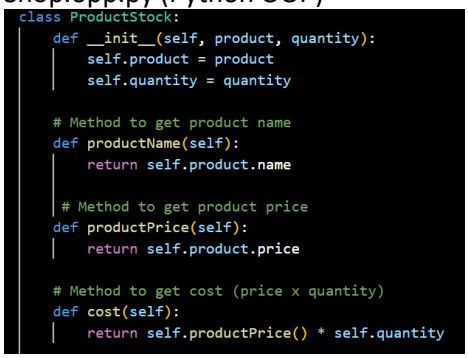
- OOP allows for inheritance, where other classes inherit features of another and extending functionality. This results in code being reusable which is not available in PP.

Shop.opp.py (Python OOP)



- The classes in OPP each contain their own methods and attributes. Since the methods are within the objects, messages can be passed from object to object. PP versions communicate by passing data as parameters to procedures.

Shop.opp.py (Python OOP)



Observations

Python and C both support PP but C does not support OOP.

PP follows a systematic approach where the execution follows an order while OOP is more dynamic.

OOP provides more functionality than PP such as inheritance and polymorphism where child and parent classes extend functionality.

OOP programming is more suited to solving real world problems and complex problems unlike PP where real world problems are harder to replicate.

Conclusion

Object oriented programming is usually less complex due to the modular nature and new objects can be created from existing ones. Objects can be reused across the program, optimising performance. The OOP code is easier to navigate as everything related to a class is contained together whereas in the procedural versions and particularly the C version, the code can be located throughout the program. Although OOP programming is more complex code to write, this is outweighed by the advantages especially when tackling real world problems, as they are difficult to replicate using PP.

References:

1. Multi-Paradigm Programming [Internet]. Available from: https://vlegalwaymayo.atu.ie/course/view.php?id=8666
2. What is Procedural Programming? A Beginner's Guide [Internet]. Available from: https://programiz.pro/resources/what-is-procedural-programming/
3. Procedural Programming Wiki [Internet]. Available from: https://en.wikipedia.org
4. Procedural Programming [definition] [Internet]. Available from: https://hackr.io/blog/procedural-programming
5. Differences between procedural and object oriented programming [Internet]. Available from: https://www.geeksforgeeks.org/differencesbetween-procedural-and-object-oriented-programming/
6. Dev.w3.org. Available from: https://dev.w3.org/libwww/Library/src/vms/getline.c
7. Object Oriented Programming Wikie [Internet]. Available from: https://en.wikipedia.org
8. Object-Oriented Programming (OOP)[Internet]. Available from: https://www.techtarget.com/searchapparchitecture/definition/object-oriented-programming-OOP
9. C return statement [Internet]. Available from: https://www.youtube.com/watch?v=HuKYb1yN7Ik
10. C programming/C reference/nonstandard/strcasecmp [Internet]. Available from: https://en.wikibooks.org/wiki/C_Programming/C_Reference/nonstandard/strcasecmp
11. Schafer C. Python OOP Tutorial 1: Classes and Instances [Internet]. Available from: https://www.youtube.com/watch?v=ZDa-Z5JzLYM
12. Python Simplified. Python Classes and Objects - OOP for Beginners [Internet]. Available from: https://www.youtube.com/watch?v=f0TrMH9s-VE
13. Python Object Oriented Programming Tutorial [Internet]. Available from: https://www.programiz.com
14. Python's Instance, Class, and Static Methods Demystified [Internet]. Available from: https://realpython.com/instance-class-and-static-methods-demystified/
15. Python as a multi paradigm programming language [Internet]. Available from: https://tracyrenee61.medium.com/python-as-a-multi-paradigm-programming-language-13b0be049a6f
16. Multi-Paradigm Languages [Internet]. Available from: https://www. oreilly.com/radar/multi-paradigm-languages/
17. 4 Advantages of Object-Oriented Programming [Internet]. Available from:https://www.roberthalf.com/us/en/insights/career-development/4-advantages-of-object-oriented-programming