[OBJECT-ORIENTED vs PROCEDURAL PROGRAMMING]

[A comparison of the solutions achieved using the procedural and the object-oriented approach]

[GMIT GALWAY-MAYO INSTITUTE OF TECHNOLOGY]
Higher Diploma in Science – Data Analytics
52960 - Multi-paradigm Programming
Lecturer: Dominic Carr

Completed by: Olga Rozhdestvina
Student Number: G00387844

December 15, 2020

Table of Contents

Introduction 2
Functionality 2
Procedural Programming 3
Object-Oriented Programming 3
Observed Similarities 3
Observed Differences 4
Summary 6
References 7

Introduction

This report contains a comparison of programming paradigms based on the solutions from my assignment on building on a shop program.

The program is completed using three programming languages and two paradigms:

- C Procedural Programming
- Python Procedural Programming
- Java Object-Oriented Programming

Procedural and Object-Oriented Programming represent fundamentally different approaches to building solutions to specific types of problems. In this respect, C language falls under procedural paradigm, Java – under Object-Oriented, whereas Python has elements of both paradigms.

Functionality

The program simulates a convenience shop with the following properties:

- Display of the shop stock and current cash.
- Reading and processing customer orders from a CSV file.
- Live mode operation where the orders are entered by a user.
- Checks for insufficient funds and stock shortages.
- The stock CSV file is updated accordingly with each successful purchase.
- Identical “user experience” of each implementation.

Procedural Programming

Procedural Oriented Programming, or **POP**, is an approach to problem-solving that relies on procedures, also known as routines or subroutines, containing a series of computational steps for execution. The paradigm separates data from procedures and treats them as two different entities. Execution of a program is based on the concept of a procedure call.

A program takes on a problem by breaking it down into smaller sub-problems or sub-procedures. This process is called functional decomposition and it is continued until the sub-procedures are simple enough to be solved.

Object-Oriented Programming

Object-Oriented programming, or **OOP**, is a programming paradigm where all computations are carried out using classes and objects. An Object-Oriented approach may use a collection of objects that are able to pass messages when they are requested for a specific information or service via call. Moreover, objects can receive and pass messages or process information in the form of data.

Observed Similarities

1. **Predefined functions // methods.**

A predefined function is typically an instruction identified by a name that is derived from a library or a registry. In POP they are called - functions, and in OOP – methods.

C

```
fprintf(stderr, ">>>> ERROR: cannot open %s: \n", filename);
```

Python

```
print(f">>>> ERROR: cannot open {filename}")
```

Java

```
System.out.println(">>>> ERROR: cannot open stock.csv");
```

2. **Modularity**

Both programming paradigms are Modular which means they utilize modular concepts to break down complex programs into several smaller program parts. Communication between them is conveyed using chaining method.

C

```
double cost = c.shoppingList[i].quantity * c.shoppingList[i].product.price;
```

Python

```
cost = i.quantity * i.product.price
```

Java

```
double cost = productStock.getCost();
```

3. **Parameter Passing.**

Parameter Passing is a mechanism for passing information to functions (POP) or methods (OOP) as parameters.

C

```
struct ProductStock listItem = {product, productQuantity};
```

Python

```
ps = ProductStock(p, quantity)
```

Java

```
ProductStock s = new ProductStock(p, quantity);
```

Observed Differences

1. **Approach.**

Procedural programming follows top down approach while Object-Oriented follows bottom up approach.

2. **Access Specifiers.**

There are no access specifiers in POP whereas there are a few different types in Object-Oriented programming, such as private, public, protected etc.

C

```
double budget;
```

Python

```
budget: float = 0.0
```

Java

```
private double budget;
```

3. **Data encapsulation**

Procedural programming does not provide data encapsulation and OOP does.

Java

```
public class Product {
    private String name;
    private double price;

    public Product(String name, double price) {
        this.name = name;
        this.price = price;
    }

    public String getName() {
        return name;
    }
```

4. **Inheritance**

Only OOP supports inheritance which allows other classes to use attributes and functions of the parent class.

Java

```
public class DiscountedProductStock extends ProductStock
```

5. Global variables

Procedural programming allows for global variables if data needs to be shared among all the functions in the program, while in OOP the same data can only be accessed through the member functions of the class.

Summary

The purpose of the report was to compare Procedural and Object-Oriented programming paradigms based on the Shop simulator.

Although these approaches have some similarities, those similarities are superficial. Moreover, there are a number of other differences between POP and OOP, like polymorphism and data abstraction, that were not mentioned in the above analysis since they weren't encountered on completion of the program. This allows me to conclude that these approaches are fundamentally different from each other.

References

1. TechDifferences: ‘Difference between OOP and POP’
   Available at: https://techdifferences.com/difference-between-oop-and-pop.html (Accessed: 14 December 2020)
2. VIVADIFFERENCES: ‘Understanding Procedural Vs. Object-Oriented Programming With Example’
   Available at: https://vivadifferences.com/difference-between-procedural-and-object-oriented-programming-with-example (Accessed: 14 December 2020)
3. Barry Allen: ‘Difference between procedural programming and object-oriented programming’, *Go Coding*
   Available at: https://gocoding.org/difference-between-procedural-programming-and-object-oriented-programming/ (Accessed: 14 December 2020)