[Date]

JAMES CONNOLLY – G00232918

Multi-Paradigm Programming

Shop

Contents

[Introduction 2](#_Toc153282789)

[Overview: 2](#_Toc153282790)

[C Program 2](#_Toc153282791)

[Object Orientated Programming 2](#_Toc153282792)

[Procedural Python Program 2](#_Toc153282793)

[Similarities – 3](#_Toc153282794)

[Differences – 5](#_Toc153282795)

[Conclusion 6](#_Toc153282796)

# Introduction

There are three programs to review that are doing a Shop simulation and the goal was to give the user the same experience using the different paradigms. The initial development was completed in our lectures throughout the semester with Dr. Dominic Carr, with templates given to begin the rest of the development. This report will analyse the three Shop simulations in the different languages and paradigms which are the following:

1. C Program
2. Object-Orientated Python Program
3. Procedural Python Program

I will give an overview of each and provide some commentary on the similarities and differences.

## Overview:

### C Program

The implementation style is procedural with functions and the user interacts with a console-based menu system. The stock and customer details are read in the program via CSV files.

### Object Orientated Programming

The implementation is OOP with the Python Language, the user interacts with the shop via a Live Shopping mode and presented with a menu. The stock and customer details are read into the program via CSV files. The program utilises classes for products.

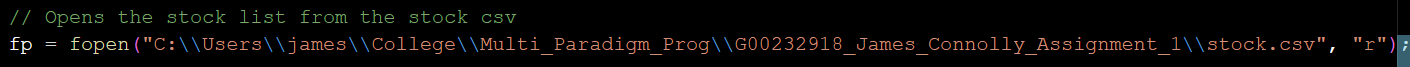
### Procedural Python Program

This program is implanted with Python obviously and it utilises the Python data classes to set up the different classes to be used in the program. This user interacts with this shop via menu and live shopping mode. Again, this program reads in the csv files for the customer and stock data.

## Similarities –

Firstly, all three programs read the customer and stock data from CSV file, which are functions that parse the data and give the results based off the requests. Although they are similar in how they get the data there is a difference between the C program and the Python programs that the paths were different, the C program required the full path, see below image which demonstrates:

C program path –



Python program path –



The menu for each program is similar and the same options are presented in each:

C program –

A screen shot of a computer

Description automatically generated

Python OOP –

A screen shot of a computer

Description automatically generated

Procedural Python Program –

A screen shot of a computer

Description automatically generated

Each program, the options do the same thing which are –

1. Show the shop’s current stock list and the float which is in the stock.csv.
2. Read in John’s Shopping list and provide the details.
3. Read in Tony’s Shopping list and provide the details.
4. Read in Bob’s Shopping list and provide the details.
5. Here the user selects 5 and can select from the stock and it calculates the details based on the choices.

This menu is consistent through the three programs and provides the user with the same experience.

## Differences –

The C program is in a more procedural style where the code is organised into to functions to handle the different task. The stock and customer data are stored in arrays reflecting this structural style approach. The live shopping mode uses a loop with the switch case that allows the code executed to changed based off the user selection in the menu.

Switch –

A screen shot of a computer program

Description automatically generated

In both the Python programs, it is more user friendly to understand as the menu choices are based on if and elif statements.

In the object orientated Python program, the classes represent the different entities but in this program the Customer class is used to read in the CSV to create the shopping list. The reason it is done this way because part of OOP is encapsulation where you want to the data and methods operating together for better organisation and readability.

The Procedural Python Program uses the more modern approach of the data classes. These classes act like templates and make the code readable.

## Conclusion

The main challenge of the project was getting familiar with the C language as this was the first time, I used it. The lecture notes proved to be a great help and inspiration for the program I created.

Inheritance was a common feature for the 3 programs as this allows the code to be more readable and reusable which is probably the main takeaway from the project. It allows you to understand the code and it encourages you to do more procedural coding when you are trying to work in a step-by-step process. It gave me a good understanding of how this type of methodology can be used in the workplace. I would also say that is important to have a plan of how you want to achieve your result, so you are not repeating information in different classes.

This project proved you can achieve similar results using different methodologies. Again, I would say it is important to choose what best suits your requirements. C allow you have control over memory which can be crucial for large datasets but on the other hand Python does the memory management automatically. For data manipulation, python has lots of libraries to choose from which allows you present the data visually. For ease of development and having experience I would choose Python.