# Multi-Paradigm Programming – Shop Assignment

## Student name: Aidan O’Connor

## Student number: G00364756

### Programming Paradigms

The term programming paradigm is the style or way of thinking about approaching problems. There are a number of different programming paradigms. The most common are:

1. Object Orientated
2. Functional
3. Imperative

### Procedural Programming

Procedural programming is based on the concept of the procedure call. Procedures contain a series of computational steps to be carried out. The sequence of these computational steps is very important in procedural programming. The C programming language is one that follows the procedural programming paradigm.

### Object-orientated Programming

Object-orientated programming is a paradigm which employs the concept of objects. Objects can be defined as data fields that have unique attributes and behaviour. The program is organised around the data rather than functions and logic. The Java programming language is one that follows the object-orientated programming paradigm.

### Differences between Procedural and Object Orientated Programming

**Order of functions and methods**

Below is a snippet of the code I developed for the C programming part of the Shop assignment. It is the main function which is called upon execution of the file when compiled and run on the machine.

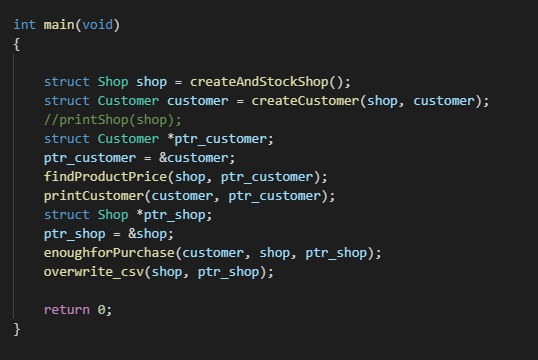


Figure : Main function, C program for Shop assignment

It is difficult to add functionality to the C program. Any additions need to be incorporated carefully as procedural code is a step by step approach. Inserting a new step in the incorrect place will have a knock-on effect to the whole program. The functions and structs must be read in the correct sequence otherwise the program will not run or will give an unintended output. I ran into problems several times in my development from defining a function before another function which relied on its input and it led to troubleshooting and reorganising the sequence of the functions.

This is not an issue in object orientated programming. Methods can be defined in any order within the program and called in the main method which will not throw up any errors. The sequence in which the methods are defined and called does not matter. Below are snippets of the Java program I developed to answer the shop assignment.



Figure : Java program, Main method Shop assignment

For example in the above main method the “enoughtoBuy()” method relies on numerous other methods defined in the class “Shop”. The method “modifyCSV” is defined after it is called in the “enoughtoBuy()” method however this does not matter in the object-oriented programming language Java as it retrieves the method wherever it is declared and runs the code.

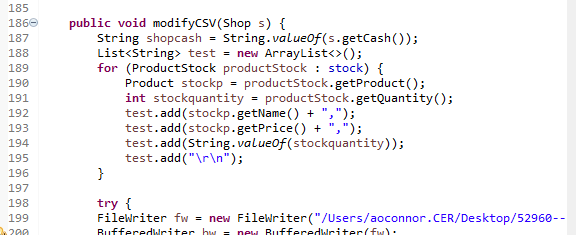


Figure : ModifyCSV method

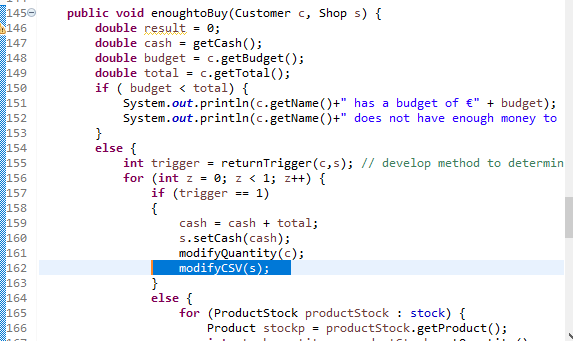


Figure : enoughtoBuy() method

**Abstraction**

C has a low level of abstraction which means certain aspects need to be managed manually by the developer rather than automatically like other programming languages with a higher level of abstraction (e.g. Java). An aspect that needs manual management is memory. This lower level of abstraction can be cumbersome for developers. I found this aspect difficult as I ran into problems often forgetting to declare variables for memory. In the code below variables needed to be declared for allocation of memory [ char \*name = malloc(sizeof(char) \* 50); ] before data could be assigned to the variables.



Figure : Function to create and stock the shop in C.

Object-oriented programming has a higher level of abstraction meaning time does not need to be given to cumbersome tasks such as memory management. This does mean efficiency losses but modern computers are so fast that the efficiency losses are not a problem for the average developer.

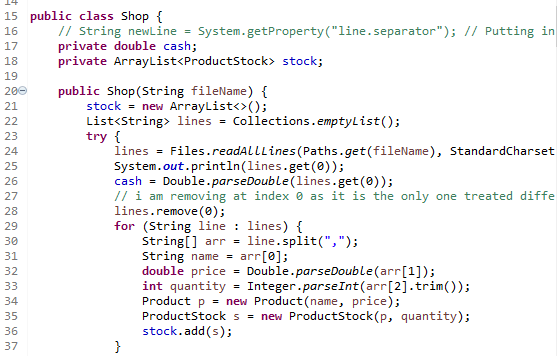


Figure : Java method for creating and stocking shop

Even though 2 different programming paradigms are employed to produce the shop assignment they can both produce the same output given that they are programmed to do so.

### Output of C program – Shop Assignment

Below are the csv inputs for my C program.

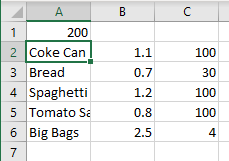
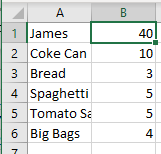


Figure : Order csv input

Figure : Stock csv input

Below is the console output to my C program for the above inputs.

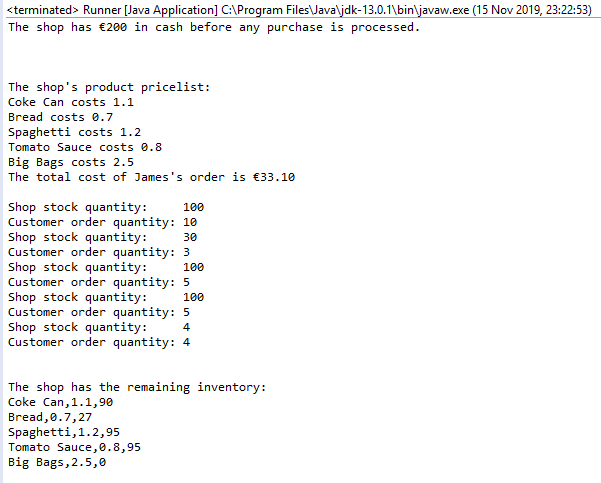


Figure : Console output for C program for the shop assignment

Below is the csv output to my C program for the above inputs. The shop’s state is consistent as the csv is overwritten.

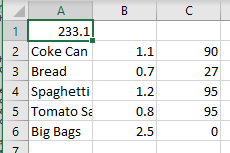
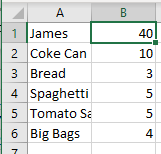


Figure : Output to Stock CSV after the C program has run

If the inputs are changed so that the customer does not have enough money to purchase the order or the shop does not have enough stock to fill the order an appropriate error message will show in the console.

### Output of Java program – Shop Assignment

Below are the csv inputs for my Java program.

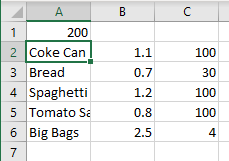
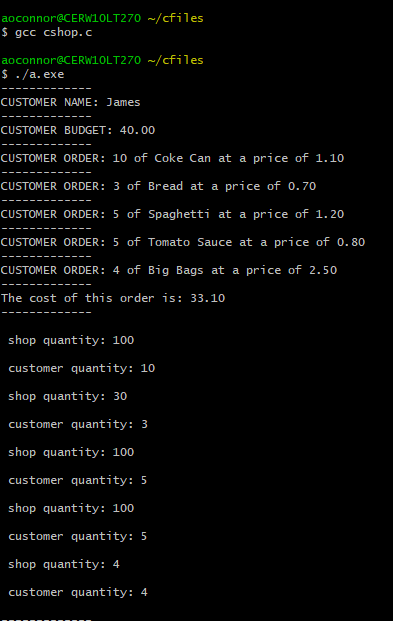


Figure : Stock csv input

Figure : Order csv input

Below is the terminal output to my Java program for the above inputs.



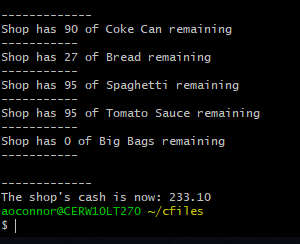


Figure : Output to terminal, Java program for Shop assignment

Below is the csv output to my Java program for the above inputs. The shop’s state is consistent as the csv is overwritten.

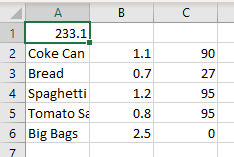


Figure : Output to Stock CSV after Java program runs

If the inputs are changed so that the customer does not have enough money to purchase the order or the shop does not have enough stock to fill the order an appropriate error message will show in the console.

# Conclusions

There are a number of programming paradigms available for developers to chose from.