# Multi-Paradigm Report

## Introduction to Programming Paradigms

Programming paradigms are a way to classify programming languages based on their features. Some languages are developed to support a single paradigm (Smalltalk and Haskell), whilst others are designed to support multiple paradigms (Ruby, Python and Java).

The two main approaches to programming are imperative programming and declarative programming.

Imperative programming is based on Von Neumann architecture and is one of the oldest paradigms. It explicitly tells the computer what to do and how to do it – a sequential set of instructions. Declarative programming focuses on what needs to be accomplished instead of a set of instructions on how to accomplish it.

The three main paradigms in the imperative programming approach are Procedural Programming Paradigm, Object Oriented Programming Paradigm and Parallel Processing Paradigm. The three paradigms in the declarative programming approach are Logic Programming Paradigm, Functional Programming Paradigm and Database Processing Paradigm.

This assignment focuses on the comparison of a procedural programming language namely C and an object-oriented programming language namely Java.

## Objective

To create a simulation of a shop in two programming languages that follow different paradigms – procedural and object-oriented. The two languages are C and Java.

This report includes a background on C and Java and outlines the main differences between the two languages using examples from the shop programs.

## Procedural Programming Paradigm - C

The C programming language was created in 1972 by a computer scientist called Dennis Ritchie at the Bell Laboratories. C is a very powerful yet low level programming language that follows a procedural programming paradigm. It is a simple, machine-independent language that provides fast execution of code.

C is a structured oriented programming language which makes it easy to test, maintain and debug. C is also a compiled language – a compiler such as MinGW compiles the source code and generates an object file (machine code). A linker links all the object files together and creates an executable file.

C is widely used in embedded systems, Adobe applications, developing browsers such as Google Chromium, developing databases such as MySQL, developing operating systems such as Microsoft Windows (desktop and mobile) and many more. It is also the “back bone” of many other programming languages such as Python, Java and C++.

## Object Oriented Programming (OOP) - Java

The Java programming language was developed by James Gosling at Sun Microsystems in 1995. Java follows an object-oriented programming paradigm which is simple, robust and secure. It was created for developers to “write once, run anywhere” – meaning it is compiled once without the need to be recompiled.

Java code is compiled into Java bytecode rather than machine code which is then executed by a Java virtual machine (VM). This use of bytecode makes porting simpler, but execution of code is slower and uses more memory.

Java code is written inside classes and all the data items are an object, except for primitive data types such as integers, floating point numbers, Boolean values and characters.

Java is used for web development, android development, desktop applications and internet of things.

## Difference between C and Java

There are many differences between C and Java, these are as follows:

* C needs to be compiled each time the code is run whereas Java code is compiled once without the need for recompilation.
* In C, memory management resides manually with the programmer, Java supports automatic memory management.
* Execution of C code is faster than Java code. This is because C is compiled into machine code whereas Java is compiled into bytecode. C is a compiled language and Java is an interpreted language.
* C uses structs which store data. Java uses classes and objects which can store data and methods.
* Java requires more memory than C.
* Java is an object-oriented programming language whereas C is a procedural programming language.
* C is a low-level language, Java is a high-level language.
* C does not contain the property called inheritance because it does not support OOPS – this is useful for code reusability. Java does contain the property inheritance.
* C supports pointers, Java does not support pointers.
* A main method in C and Java acts as the entry point of execution of the program.
* C does not have any proper way for hiding data, so it is less secure. Java provides data hiding so it is more secure.

## Shop Program in C:

* Firstly, four entities were defined in C using structs (structures). The four structs defined were Shop, Customer, Product and ProductStock. These structs contained varying data types.
* Re-usable void methods were created to access the data in the structs and to print the product and customer information. These methods were called through the “main()” method and the information was printed to the console.
* In C the number of elements in an array is unknown so an index was created to keep track of the values by storing them in an index.
* Reading in data from csv files is quite complex in C – a lot of code is required to create the method to read in the csv file. Much simpler in high-level languages such as Java.
* New memory had to be manually allocated in C when reading in the stock.csv file. This was needed to store each product’s name. The quantity and price had enough memory as we used atoi and atof which created more memory.
* New memory had to be manually added when reading in the customer.csv file too.

## Shop Program in Java:

* Firstly, four classes were created in Eclipse. Theses four classes were Shop, Customer, Product and ProductStock.
* We created character variables in each of the four classes and created constructors as well. In Java, a constructor is a method. It is called when an instance of the class is created. Data and methods can be stored in classes. In C only data can be stored in the struct.
* Used array lists rather than low level arrays as we did with the program in C. No need to keep track of the index.
* Variables within the class were set as private which means they can not be accessed outside of the class. This makes the program more secure.
* The toString() method returns the string representation of the object. These were created in each class.
* It was a lot easier to read the csv files in Java.
* A main method was created – this is where the execution of the code begins.

## References

<https://www.geeksforgeeks.org/introduction-of-programming-paradigms/>

<https://en.wikipedia.org/wiki/Programming_paradigm>

<https://www.guru99.com/c-programming-language.html>

<https://en.wikipedia.org/wiki/Java_(programming_language)>

<https://theailearner.com/2018/09/26/programming-paradigms/>

<https://www.coursereport.com/blog/what-is-java-programming-used-for>

<https://www.computerhope.com/jargon/i/imp-programming.htm>

<https://www.geeksforgeeks.org/differences-between-procedural-and-object-oriented-programming/>