**You must complete a short report, around 3-5 pages, which compares the solutions achieved using the procedural approach and the object oriented approach.**

**The stock CSV should hold the initial cash value for the shop.**

When comparing the difference between programming in C and programming in Python, it possibly helps to know that the C programming language was created in 1972, that it precedes Python which was created in 1991. This is what the programming community say about C and Python; C is not object oriented language. C is a general-purpose, imperative language, supporting structured programming. [[1]](#footnote-1) Python supports both Object Oriented and Procedural Programming language as it is a high level programming language designed for general purpose programming.[[2]](#footnote-2)

One of the first tasks for this assignment was to bring in the data from the **stock CSV that should hold the initial cash value for shop**. Let’s look at the structured programming approach of C.

*Structure is a user-defined datatype in C language which allows us to combine data of different types together. Structure helps to construct a complex data type which is more meaningful. It is somewhat similar to an Array, but an array holds data of similar type only. But structure on the other hand, can store data of any type, which is practical more useful.[[3]](#footnote-3)*

struct keyword is used to define a structure. struct defines a new data type which is a collection of primary and derived datatypes.

**Syntax:**

struct [structure\_tag]

{

//member variable 1

//member variable 2

//member variable 3

...

}[structure\_variables];

**C**

|  |
| --- |
| struct Shop {  double cash;  struct ProductStock stock[MAX\_PRODUCTS\_IN\_STOCK];  int index;  }; |

Here struct Shop declares a structure to hold the details of a shop which consists of 3 data fields, namely cash, ProductStock, and index. These fields are called structure elements or members.

Each member can have different datatype, like in this case, cash is a double (in Python a double is called a float), ProductStock is a struct referencing the stock element of another struct called ProductStock, and index is an integer.

Python Procedural

As many Python programmers may have formerly been C programmers, there will be times when a python programmer, may want the same capability as Structs that are used in C.

In programming the procedural version of Python, to create a collection of primary and derived datatypes to be used in the same way a C programmer would use Structs you would import the module dataclasses.

**from dataclasses import dataclass, field. Note the terminology data classes is used.** What makes this a data class is the [@dataclass decorator](https://realpython.com/primer-on-python-decorators/) just above the class definition. Beneath the @dataclass you simply list the fields you want in your data class.

The notation used for the fields is using a new feature in Python 3.6 called [variable annotations](https://www.python.org/dev/peps/pep-0526/).

https://realpython.com/python-data-classes/

**Procedural Style**

**The impo**

|  |
| --- |
| @dataclass  class Product:  name: str  price: float = 0.0  @dataclass  class ProductStock:  product: Product  quantity: int  @dataclass  class Shop:  cash: float = 0.0  stock: List[ProductStock] = field(default\_factory=list) |

Python Object Orientated

**Read in customer orders from a CSV file.**

<https://www.programiz.com/c-programming/c-file-input-output>

<https://www.programiz.com/c-programming/examples/read-file>

<https://www.tutorialspoint.com/cprogramming/c_strings.htm>

**The shop must be able to process the orders of the customer.**

C Procedural Method

This function reads the customer’s CSV file and process each of the ordered product in following steps:

1. First of all, it checks if the ordered product exists in shop’s stock or not
2. If the product exists, it checks if there is sufficient quantity of that product available in stock in order to meet consumer’s required quantity.
3. It then fetches the product’s price from the shop’s stock
4. It multiplies the product price with the ordered quantity to get the total cost of that product
5. After going through each of the products in consumer’s order, it then checks if the grand total of the consumer’s order is within consumer’s budget
6. If the order is within budget, the order is processed. The respective ordered products’ quantities are removed from the stock and the order’s total value is added in the shop’s total cash value.

**Update the cash in the shop based on money received.**

C Procedural Method

If the consumer order’s total value is within consumer’s budget, then the order is processed and the total amount of consumer’s order is added to shop’s cash, and a success message is shown on screen. Otherwise, a fail message is displayed and no amount is added to the shop’s cash value.

**It is important that the state of the shop be consistent.**

C Procedural Method

When the consumer’s order is processed, the shop’s stock of each product is updated accordingly to ensure that the state of the shop remains consistent after each order.

**– Know whether or not the shop can fill an order.**

C Procedural Method

The first code checks if the ordered product exists in the stock or not

The second code checks if the ordered product’s quantity is available in stock or not

**\* Thrown an appropriate error.**

C Procedural Method

The first error message is thrown in “main” function, when the user enters an incorrect choice

The second error message is thrown in the “printCustomer” function when entered product doesn’t exist in stock

The third error is thrown at several positions in the code, whenever the code fails to open an input csv file. This is in fact a program exit code. More details on: <https://en.cppreference.com/w/c/program/EXIT_status>

**Operate in a live mode, where the user can enter a product by name, specify a quantity, and pay for it. The user should be able to buy many products in this way.**

C Procedural Method

Taking Input and Printing Output:

<https://www.programiz.com/c-programming/c-input-output>

Structures:

[https://www.studytonight.com/c/structures-in-c.php#](https://www.studytonight.com/c/structures-in-c.php)

Pointers to structures:

<https://www.programiz.com/c-programming/c-structures-pointers>

1. https://stackoverflow.com/questions/3241932/is-the-c-programming-language-object-oriented [↑](#footnote-ref-1)
2. https://www.tutorialspoint.com/is-python-object-oriented-or-procedural [↑](#footnote-ref-2)
3. https://www.studytonight.com/c/structures-in-c.php [↑](#footnote-ref-3)