Python Project

Invoice generator

By: Aaron Tamang

Table of Contents

Table of Figures

Table of Tables

1. Introduction:

The goal of this project is to develop a program that manages the inventory of a laptop shop that buys and sells laptops from manufacturers and customers, respectively. The program will allow the laptop shop to keep track of the available laptops and update the stock based on the transactions made with customers and manufacturers.

The program will be designed to read a text file containing information about the available laptops, including their names, brands, prices, quantities, processors, and graphics cards. With each transaction, the program will update the text file and generate a note or receipt with the details of the transaction.

The objectives of this project are to:

- To develop a program that is capable of reading and manipulating data from a text file using Python's data structures and operations. Python is a popular programming language used for data manipulation and analysis, and it offers a wide range of built-in functions and modules that make it easy to read and manipulate data from various sources.
- To implement the program in a modular way, with separate functions for input/output, reading files, generating invoices/notes, and updating the stock. This approach will make the program easier to understand, test, and maintain. The program will have a user-friendly interface, making it easy for the laptop shop staff to use.
- Generate notes or receipts for each transaction, including information such as the name of the laptop, name of the brand, date and time of purchase, total amount without shipping cost, shipping cost, and total amount to be paid for the laptops.
- To ensure that the stock of each laptop is updated accurately and in real-time, regardless of whether the laptop is sold to a customer or purchased from a manufacturer. To achieve this objective, the program must be designed to constantly update the stock based on the transactions made with customers and manufacturers.

In conclusion, this project aims to develop a program that can efficiently manage the inventory of a laptop shop, generate accurate notes or receipts, and update the stock in real-time. The program will be implemented in a modular way, with a user-friendly interface, making it easy for the laptop shop staff to use. The success of the project will be determined by its ability to streamline the inventory management process, reducing the risk of overstocking or understocking, and providing an improved customer experience.

2. Discussion and Analysis:

2.1 Algorithm:

Step 1: Call the function ReadingLaptopText() from the read module and return the value to laptop_dict.

Step 2: Prompt the user to choose between ordering from the manufacturer or selling to the customer.

Step 3: If the user chooses to order from the manufacturer, prompt them to enter the name of the laptop and the quantity to be ordered.

Step 4: Generate an order invoice with the necessary details.

Step 5: Update the stock of the laptops in the text file by adding the quantity ordered.

Step 6: If the user chooses to sell to the customer, prompt them to enter the name of the laptop, the quantity to be sold, and the name of the customer and phone number.

Step 7: Generate a sales invoice with the necessary details.

Step 8: Update the stock of the laptops in the text file by subtracting the quantity sold.

Step 9: Repeat steps 2-8 until the user chooses to exit the program. This has already been handled in the loop structure starting from Step 2.

Step 10: Save all changes made to the text file.

2.2 Flow Chart:

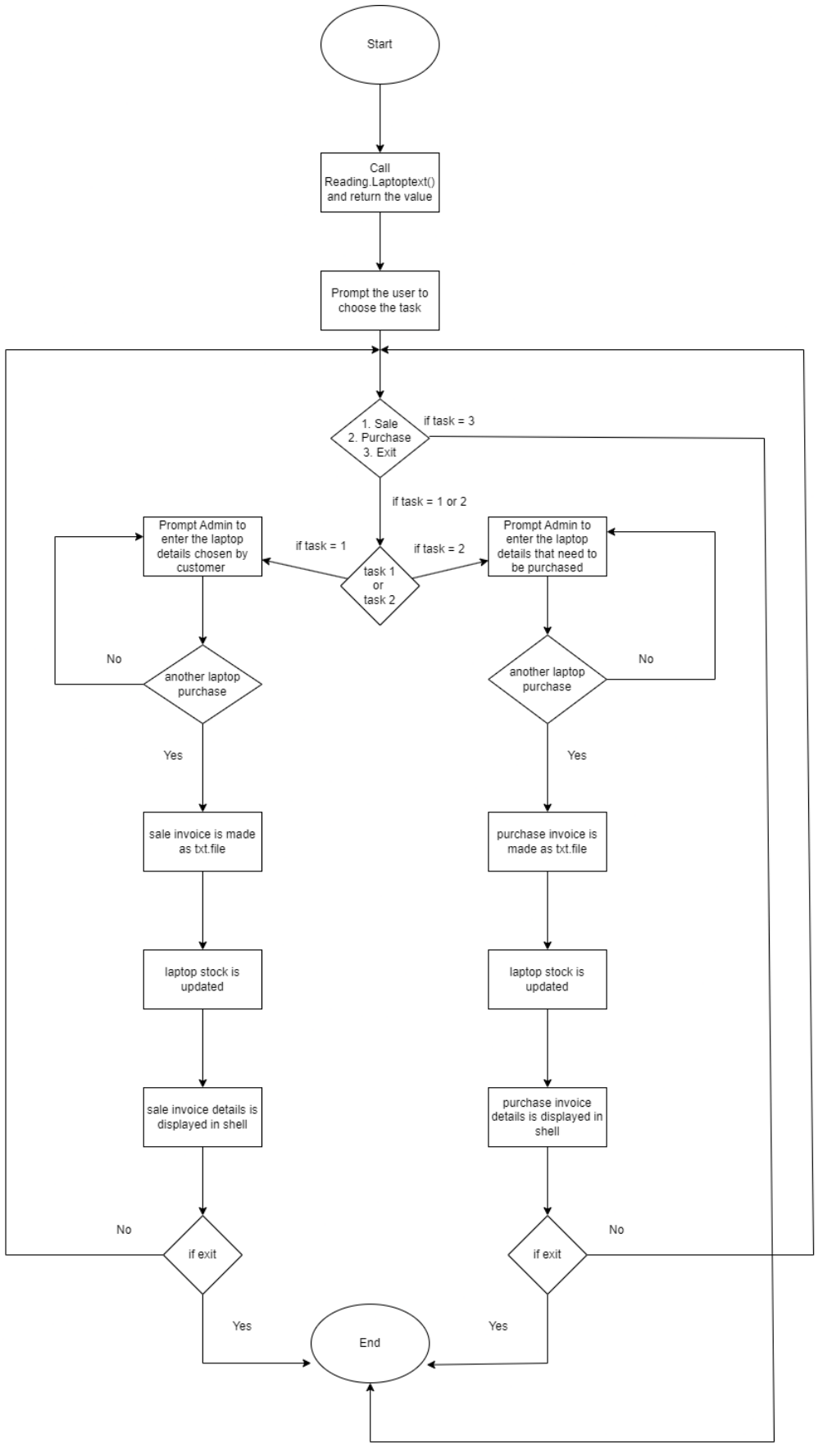

Figure 1: Flowchart of program.

2.3 Data Structures:

- String:

```
name = input("Enter the name of the customer: ")
while not name:
    print()
    print("Name cannot be empty.\n")
    name = input("Enter the name of the customer: ")
```

Figure 2: String variable.

A string is a built-in data type in Python used to represent sequences of characters. It handles textual data and is an immutable sequence of Unicode code points. Strings are simple to create and easy to use in Python. The input () function is used to receive input from the user. This function always returns the input as a string, regardless of whether the user enters numbers or text.

- List:

```
boughtlaptops= []

continueValue = True
while continueValue:
    read.available_laptops()
    laptop_data, price, quantity, current_quantity, laptop_choice = operation.get_laptop_details(laptop_dict

    #time = datetime.datetime(.now()
    BrandName = laptop_data[1]
    laptopName = laptop_data[0]
    userQuantiy = quantity
    unitPrice = int(price)
    totalPrice = int(unitPrice) * int(userQuantiy)
    boughtlaptops.append((BrandName,laptopName,userQuantiy,f"${unitPrice}",f"${totalPrice}"))
    invoice_sale = f"Sold_{time.strftime('%Y %B %d_%H-%M-%S')}.txt"
```

Figure 3: List variable.

Lists are mutable, which means Python will not create a new list if we modify an element of the list. To create a Python list, we place all the items or elements inside square brackets [ ], separated by commas **,**. All the elements in the list are stored in the index basis with starting index 0.A list can have any number of items, and they may be of different types, such as integer, float, string, etc. boughtlaptops = [] is an empty list created to store the details of the laptops that a user has purchased.The append() method is used to add a new element at the end of the list. In this case, a tuple is being appended to the list.

- Tuple:

```
BrandName = laptop_data[1]
laptopName = laptop_data[0]
userQuantiy = quantity
unitPrice = int(price)
totalPrice = int(unitPrice) * int(userQuantiy)
boughtlaptops.append((BrandName,laptopName,userQuantiy,f"${unitPrice}",f"${totalPrice}"))
```

Figure 4: Tuple variable.

A tuple in Python is like a list. The only difference is that a list is enclosed between square brackets [ ], while a tuple is enclosed between parentheses (). Tuples are immutable, which means we cannot change the elements of a tuple once it is assigned. The append () method is used to add a new element at the end of the list. In this case, a tuple is being appended to the list. The tuple contains the details of a purchased laptop: BrandName, laptopName, userQuantity, unitPrice, and totalPrice.

- Dictionary:

```
def ReadingLaptopText():
    with open("Laptops.txt","r") as file:
        laptop_dict= {}
        num = 1
        for line in file:
            line = line.replace("\n","")
            laptop_dict.update({num:line.split(",")})
            num = num + 1
    return laptop_dict
```

Figure 5: Dictionary variable.

A dictionary is a mutable data structure, where a value can be updated. The keys in a dictionary must be unique and immutable, while the values are accessed using the corresponding key. The values in a dictionary can be updated, while the keys cannot be changed once set. Due to the key-value pairing and index-like behavior based on keys, a dictionary is often referred to as an associative array. The laptop_dict dictionary is created with an empty dictionary {}. Then, the function opens the "Laptops.txt" file in read mode using the open () function with the "r" parameter. It reads the lines of the file using a for loop and updates the laptop_dict dictionary with the laptop details. The update () function is used to add the new key-value pair to the laptop_dict dictionary. The key is represented by the num variable which is incremented in each iteration, and the value is the list of laptop details.

3. Program working mechanism:

1. When the program runs, it will display the shop name and details at the top, followed by a list of tasks for the user to select from.

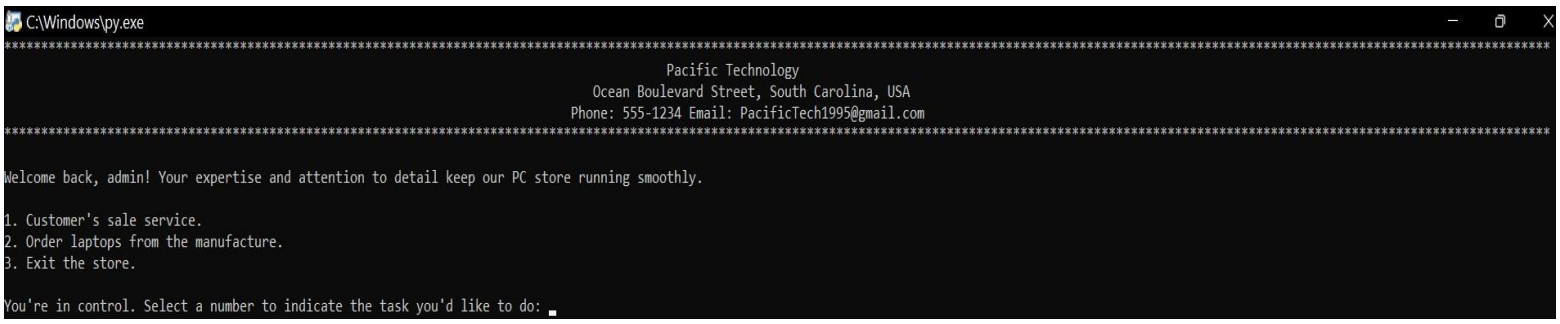

Figure 6: Initial output of the program.

2. When option 1 is selected:

- The program will initiate the process for selling laptops to customers. It will prompt the user to enter the customer's name and phone number, and then display the available laptop information.

- The customer will be asked to provide the serial number (S.N.) of the laptop they wish to purchase and specify the quantity they want to buy.

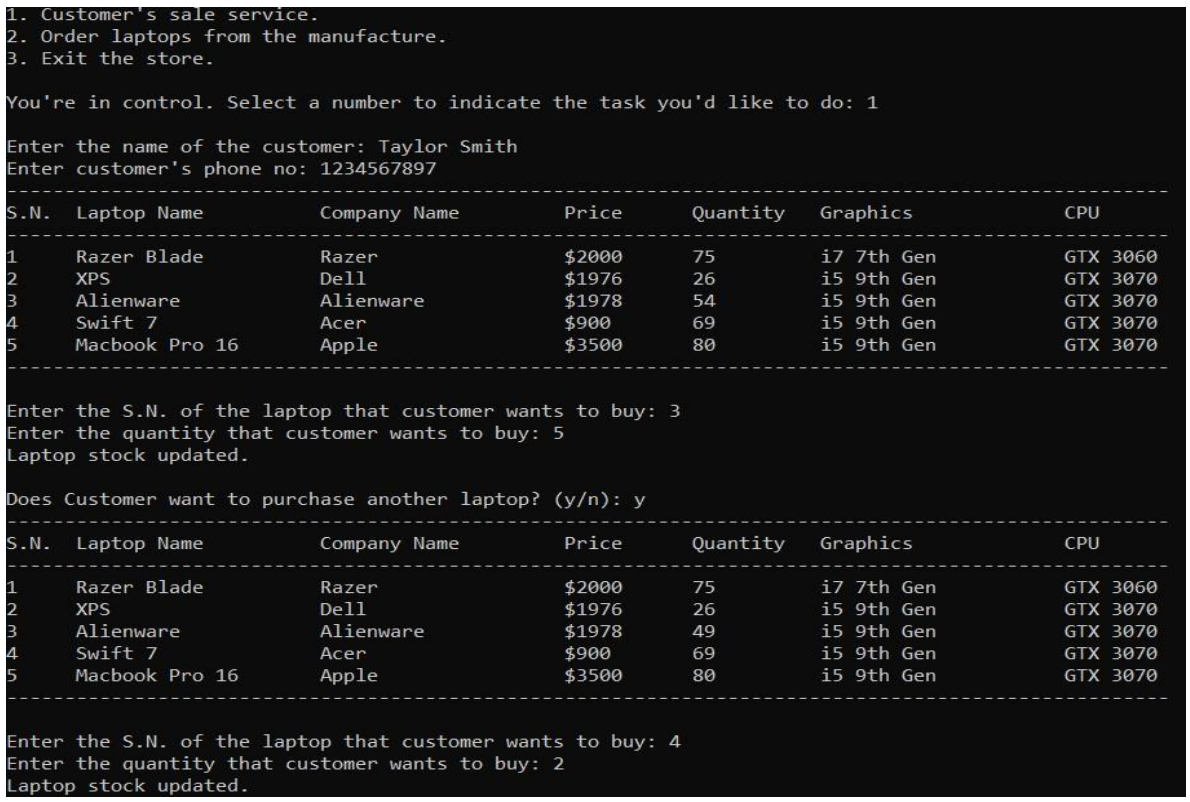

Figure 7: Execution of option 1.

- The program will then ask the customer if they wish to buy another laptop. If “y” is entered, the laptop information will be displayed again for the customer to choose from. If “n” is entered, the program will generate an invoice and display the invoice details on the screen.

```
Does Customer want to purchase another laptop? (y/n): n
-------------------------------------------------------------------------------------------
                                    Pacific Technology
                            Ocean Boulevard Street, South Carolina, USA
                          Phone: 555-1234 Email: PacificTech1995@gmail.com
-------------------------------------------------------------------------------------------

Customer Name: Taylor Smith
Phone no: 1234567897
Sold time: 2023 May 07_10-26-00

-------------------------------------------------------------------------------------------
Laptop Brand        Item Name           Total Quantity      Unit Price          Total
-------------------------------------------------------------------------------------------
Alienware           Alienware           5                   $1978               $9890
Acer                Swift 7             2                   $900                $1800
-------------------------------------------------------------------------------------------

Shipping cost: $100
Grand Total: $2978
-------------------------------------------------------------------------------------------
```

Figure 8: Invoice details.

- An invoice will be created and saved as a text file.

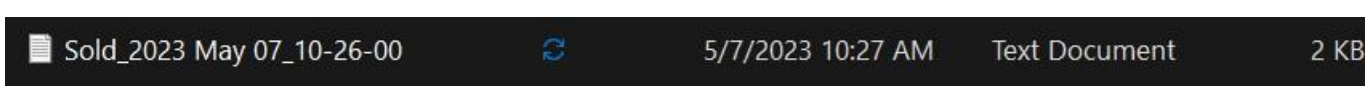

Figure 9: Invoice saved as text file.

- The invoice text, generated after selling the laptops.

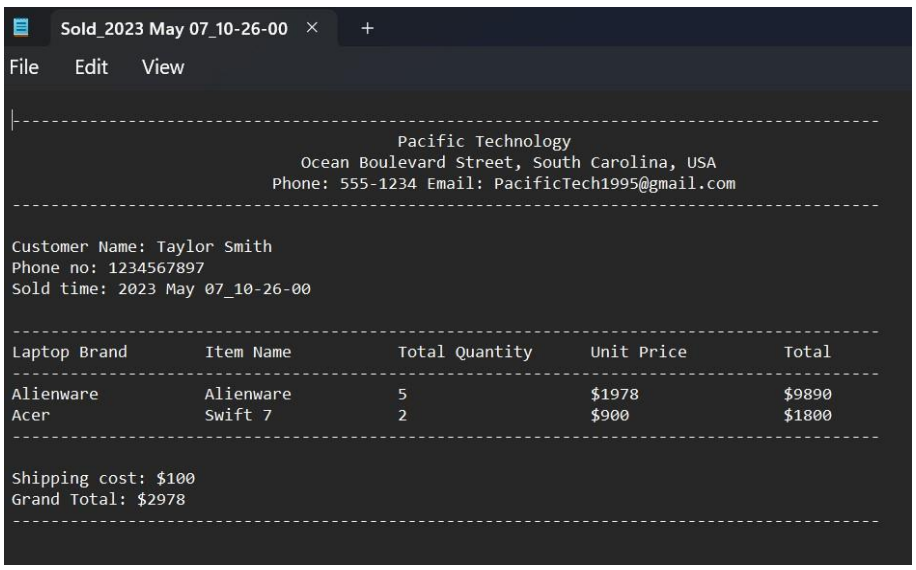

Figure 10: Invoice as text file.

3. When option 2 is selected:

- The program will initiate the process for purchasing laptops from the manufacturers. It will prompt the admin to enter their name and then display the available laptop information.

- The admin will be asked to provide the serial number (S.N.) of the laptop they wish to order and specify the quantity they want to purchase.

```
1. Customer's sale service.
2. Order laptops from the manufacture.
3. Exit the store.

You're in control. Select a number to indicate the task you'd like to do: 2

Enter your name admin: Elon Musk
-------------------------------------------------------------------------------------------------------
S.N.  Laptop Name          Company Name         Price      Quantity   Graphics            CPU
-------------------------------------------------------------------------------------------------------
1     Razer Blade          Razer                $2000      70         i7 7th Gen          GTX 3060
2     XPS                  Dell                 $1976      16         i5 9th Gen          GTX 3070
3     Alienware            Alienware            $1978      54         i5 9th Gen          GTX 3070
4     Swift 7              Acer                 $900       69         i5 9th Gen          GTX 3070
5     Macbook Pro 16       Apple                $3500      80         i5 9th Gen          GTX 3070
-------------------------------------------------------------------------------------------------------

Enter the S.N. of the laptop you want to purchase from the manufacturer: 1
Enter the quantity you want to purchase: 5
Laptop stock updated.

Admin, do you want to order another laptop? (y/n): y
-------------------------------------------------------------------------------------------------------
S.N.  Laptop Name          Company Name         Price      Quantity   Graphics            CPU
-------------------------------------------------------------------------------------------------------
1     Razer Blade          Razer                $2000      75         i7 7th Gen          GTX 3060
2     XPS                  Dell                 $1976      16         i5 9th Gen          GTX 3070
3     Alienware            Alienware            $1978      54         i5 9th Gen          GTX 3070
4     Swift 7              Acer                 $900       69         i5 9th Gen          GTX 3070
5     Macbook Pro 16       Apple                $3500      80         i5 9th Gen          GTX 3070
-------------------------------------------------------------------------------------------------------

Enter the S.N. of the laptop you want to purchase from the manufacturer: 2
Enter the quantity you want to purchase: 10
Laptop stock updated.
```

Figure 11: Execution of option 2.

- The program will then ask the admin if they wish to purchase another laptop. If “y” is entered, the laptop information will be displayed again for further selection. If “n” is entered, the program will generate an invoice and display the invoice details on the screen.

```
Admin, do you want to order another laptop? (y/n): n
-------------------------------------------------------------------------------------------------------
                                        Pacific Technology
                                Ocean Boulevard Street, South Carolina, USA
                              Phone: 555-1234 Email: PacificTech1995@gmail.com
-------------------------------------------------------------------------------------------------------

Admin Name: Elon Musk
Ordered time: 2023 May 07_10-23-24

-------------------------------------------------------------------------------------------------------
Laptop Brand         Item Name            Total Quantity       Unit Price          Total
-------------------------------------------------------------------------------------------------------
Razer                Razer Blade          5                    $2000               $10000
Dell                 XPS                  10                   $1976               $19760
-------------------------------------------------------------------------------------------------------

Total amount: $3976
VAT amount (13%): $516.88
Gross Amount: $4492.88
Admin Elon Musk authorized this transaction.
-------------------------------------------------------------------------------------------------------
```

Figure 12: Invoice details.

- An invoice will be created and saved in a text file.

Figure 13: Invoice saved as text file.

- The invoice text, generated after purchasing the laptops from the manufacturers, will be saved in a text file.

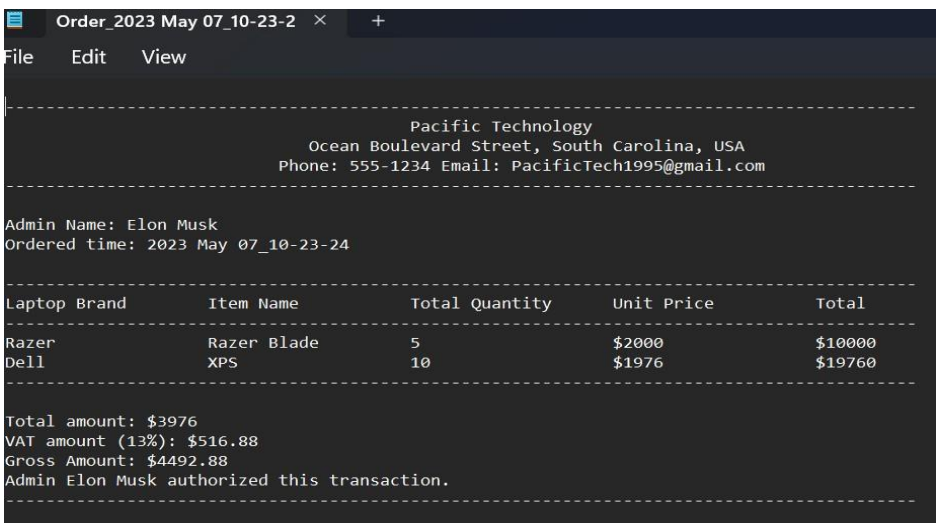

Figure 14: Invoice as text file.

4. When option 3 is selected:

- The program will terminate with a message of appreciation.

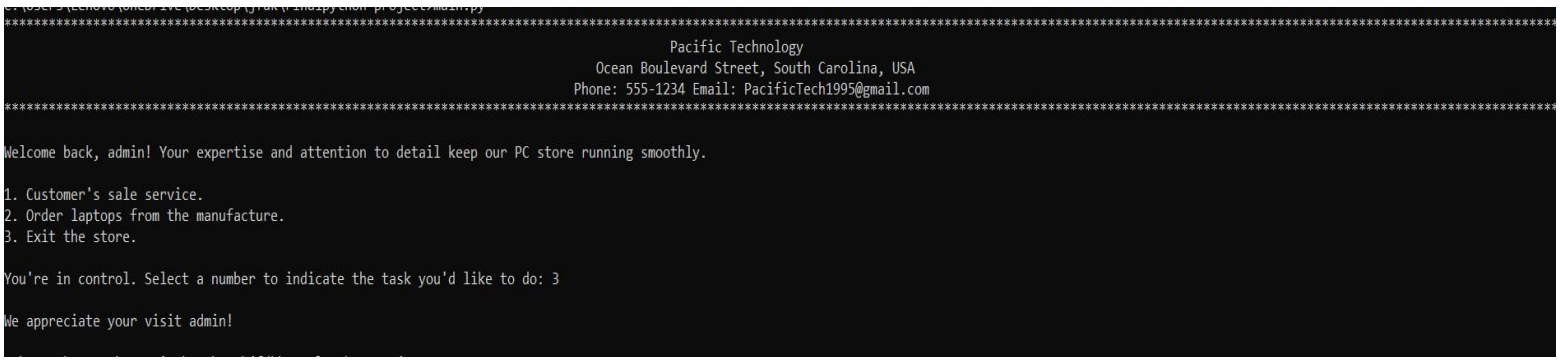

Figure 15: Execution of option 3.

4. Testing:

4.1 Test 1: Handling of expectation error.

```
You're in control. Select a number to indicate the task you'd like to do: 1

-------------------------------------------------------------------------------------------------------
S.N.  Laptop Name          Company Name         Price      Quantity   Graphics            CPU
-------------------------------------------------------------------------------------------------------
1     Razer Blade          Razer                $2000.0    161        i7 7th Gen          GTX 3060
2     XPS                  Dell                 $1976.0    52         i5 9th Gen          GTX 3070
3     Alienware            Alienware            $1978.0    59         i5 9th Gen          GTX 3070
4     Swift 7              Acer                 $900.0     69         i5 9th Gen          GTX 3070
5     Macbook Pro 16       Apple                $3500.0    80         i5 9th Gen          GTX 3070
-------------------------------------------------------------------------------------------------------

Enter the name of the customer: abc
Enter customer's phone no: 1010101010
Enter the S.N. of the laptop that customer wants to buy: abc

We're sorry, but the S.N you entered is not valid. Please enter a numeric value.

Enter the S.N. of the laptop that customer wants to buy:
```

Figure 16: Exception error indication.

Objective	Showing exception message
Action	Entering non-numeric value
Expected Result	Showing exception message
Actual Result	Exception message was shown
Conclusion	Successful

Table 1: Test 1

4.2 Test 2: Handling non-existed value.

```
1. Customer's sale service.
2. Order laptops from the manufacture.
3. Exit the store.

You're in control. Select a number to indicate the task you'd like to do: 1

Enter the name of the customer: Elon Musk
Enter customer's phone no: 1234567897
-------------------------------------------------------------------------------------------------
S.N.  Laptop Name          Company Name         Price      Quantity   Graphics            CPU
-------------------------------------------------------------------------------------------------
1     Razer Blade          Razer                $2000      70         i7 7th Gen          GTX 3060
2     XPS                  Dell                 $1976      6          i5 9th Gen          GTX 3070
3     Alienware            Alienware            $1978      56         i5 9th Gen          GTX 3070
4     Swift 7              Acer                 $900       69         i5 9th Gen          GTX 3070
5     Macbook Pro 16       Apple                $3500      80         i5 9th Gen          GTX 3070
-------------------------------------------------------------------------------------------------

Enter the S.N. of the laptop that customer wants to buy: -1

Invalid S.N. Please enter a valid S.N.

Enter the S.N. of the laptop that customer wants to buy: 1000

Invalid S.N. Please enter a valid S.N.

Enter the S.N. of the laptop that customer wants to buy:
```

Figure 17: Figure 3: Valid S.N input.

Objective	To inform the admin to enter a valid value
Action	Entering negative and non-existed value as input
Expected Result	To show the message to admin to enter a valid value.
Actual Result	Message was shown
Conclusion	Successful

Table 2: Test 2

4.3 Test 3: Complete procedure of ordering of laptop from manufacture with invoice generation.

```
1. Customer's sale service.
2. Order laptops from the manufacture.
3. Exit the store.

You're in control. Select a number to indicate the task you'd like to do: 2

Enter your name admin: Elon Musk
-------------------------------------------------------------------------------------------------------
S.N.  Laptop Name          Company Name         Price      Quantity   Graphics            CPU
-------------------------------------------------------------------------------------------------------
1     Razer Blade          Razer                $2000      70         i7 7th Gen          GTX 3060
2     XPS                  Dell                 $1976      16         i5 9th Gen          GTX 3070
3     Alienware            Alienware            $1978      54         i5 9th Gen          GTX 3070
4     Swift 7              Acer                 $900       69         i5 9th Gen          GTX 3070
5     Macbook Pro 16       Apple                $3500      80         i5 9th Gen          GTX 3070
-------------------------------------------------------------------------------------------------------

Enter the S.N. of the laptop you want to purchase from the manufacturer: 1
Enter the quantity you want to purchase: 5
Laptop stock updated.

Admin, do you want to order another laptop? (y/n): y
-------------------------------------------------------------------------------------------------------
S.N.  Laptop Name          Company Name         Price      Quantity   Graphics            CPU
-------------------------------------------------------------------------------------------------------
1     Razer Blade          Razer                $2000      75         i7 7th Gen          GTX 3060
2     XPS                  Dell                 $1976      16         i5 9th Gen          GTX 3070
3     Alienware            Alienware            $1978      54         i5 9th Gen          GTX 3070
4     Swift 7              Acer                 $900       69         i5 9th Gen          GTX 3070
5     Macbook Pro 16       Apple                $3500      80         i5 9th Gen          GTX 3070
-------------------------------------------------------------------------------------------------------

Enter the S.N. of the laptop you want to purchase from the manufacturer: 2
Enter the quantity you want to purchase: 10
Laptop stock updated.
```

Figure 18: Required details for the purchase.

```
Admin, do you want to order another laptop? (y/n): n
-------------------------------------------------------------------------------------------
                                    Pacific Technology
                              Ocean Boulevard Street, South Carolina, USA
                          Phone: 555-1234 Email: PacificTech1995@gmail.com
-------------------------------------------------------------------------------------------

Admin Name: Elon Musk
Ordered time: 2023 May 07_10-23-24

-------------------------------------------------------------------------------------------
Laptop Brand        Item Name           Total Quantity      Unit Price          Total
-------------------------------------------------------------------------------------------
Razer               Razer Blade         5                   $2000               $10000
Dell                XPS                 10                  $1976               $19760
-------------------------------------------------------------------------------------------

Total amount: $3976
VAT amount (13%): $516.88
Gross Amount: $4492.88
Admin Elon Musk authorized this transaction.
-------------------------------------------------------------------------------------------
```

Figure 19: Generated invoice of purchase.

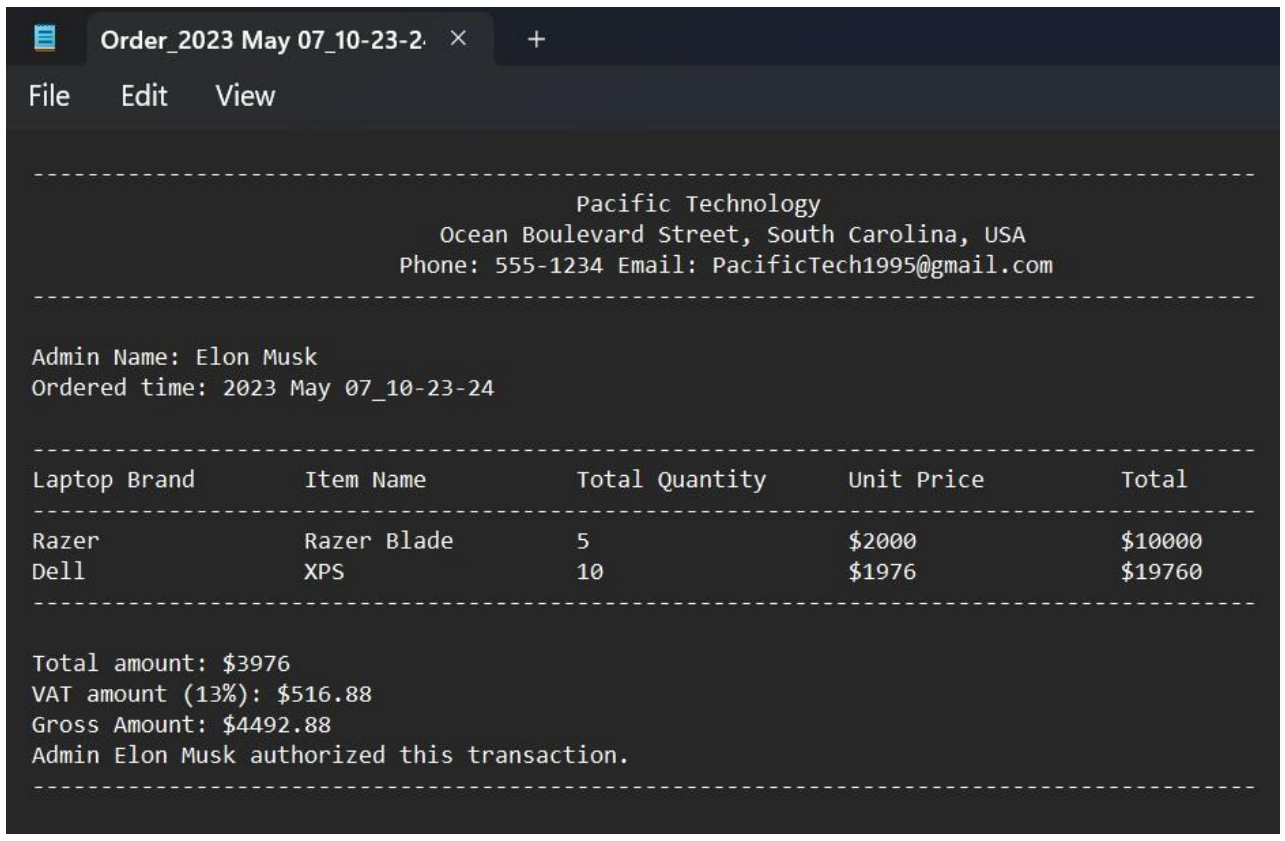

Figure 20: Invoice saved as txt file.

Objective	To show the complete purchase process and generating purchase invoice
Action	Entering the task which perform the purchase and generates its invoice
Expected Result	Purchase task should work and should generate the purchase invoice
Actual Result	Purchase task worked and purchase invoice was generated
Conclusion	Successful

Table 3: Test 3

4.4 Test 4: Complete procedure of selling laptop to customer with invoice generation.

```
1. Customer's sale service.
2. Order laptops from the manufacture.
3. Exit the store.

You're in control. Select a number to indicate the task you'd like to do: 1

Enter the name of the customer: Taylor Smith
Enter customer's phone no: 1234567897
-------------------------------------------------------------------------------------------------------
S.N.  Laptop Name          Company Name         Price      Quantity   Graphics            CPU
-------------------------------------------------------------------------------------------------------
1     Razer Blade          Razer                $2000      75         i7 7th Gen          GTX 3060
2     XPS                  Dell                 $1976      26         i5 9th Gen          GTX 3070
3     Alienware            Alienware            $1978      54         i5 9th Gen          GTX 3070
4     Swift 7              Acer                 $900       69         i5 9th Gen          GTX 3070
5     Macbook Pro 16       Apple                $3500      80         i5 9th Gen          GTX 3070
-------------------------------------------------------------------------------------------------------

Enter the S.N. of the laptop that customer wants to buy: 3
Enter the quantity that customer wants to buy: 5
Laptop stock updated.

Does Customer want to purchase another laptop? (y/n): y
-------------------------------------------------------------------------------------------------------
S.N.  Laptop Name          Company Name         Price      Quantity   Graphics            CPU
-------------------------------------------------------------------------------------------------------
1     Razer Blade          Razer                $2000      75         i7 7th Gen          GTX 3060
2     XPS                  Dell                 $1976      26         i5 9th Gen          GTX 3070
3     Alienware            Alienware            $1978      49         i5 9th Gen          GTX 3070
4     Swift 7              Acer                 $900       69         i5 9th Gen          GTX 3070
5     Macbook Pro 16       Apple                $3500      80         i5 9th Gen          GTX 3070
-------------------------------------------------------------------------------------------------------

Enter the S.N. of the laptop that customer wants to buy: 4
Enter the quantity that customer wants to buy: 2
Laptop stock updated.
```

Figure 21: Required details for the sales.

```
Does Customer want to purchase another laptop? (y/n): n
-------------------------------------------------------------------------------------------
                                    Pacific Technology
                              Ocean Boulevard Street, South Carolina, USA
                           Phone: 555-1234 Email: PacificTech1995@gmail.com
-------------------------------------------------------------------------------------------

Customer Name: Taylor Smith
Phone no: 1234567897
Sold time: 2023 May 07_10-26-00

-------------------------------------------------------------------------------------------
Laptop Brand        Item Name           Total Quantity      Unit Price          Total
-------------------------------------------------------------------------------------------
Alienware           Alienware           5                   $1978               $9890
Acer                Swift 7             2                   $900                $1800
-------------------------------------------------------------------------------------------

Shipping cost: $100
Grand Total: $2978
-------------------------------------------------------------------------------------------
```

Figure 22: Generated invoice of sales.

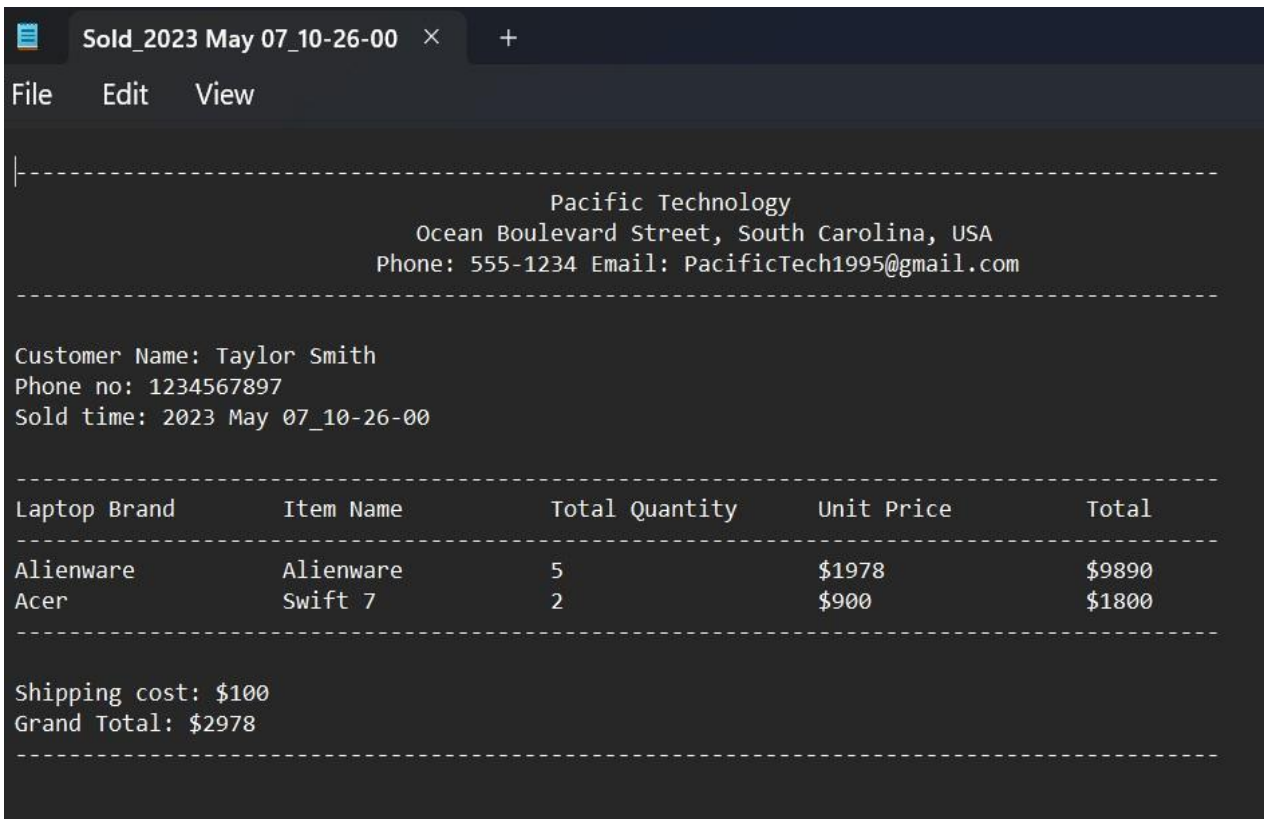

Figure 23: Invoice saved as txt file.

Objective	To show the complete sale process and generating sale invoice
Action	Entering the task which perform the sale and generates its invoice
Expected Result	Sale task should work, and it should generate the sale invoice
Actual Result	Sale task worked and sale invoice was generated
Conclusion	Successful

Table 4: Test 4

4.5 Test 5: Updates of the product after sales and purchase.

- Sales

```
1. Customer's sale service.
2. Order laptops from the manufacture.
3. Exit the store.

You're in control. Select a number to indicate the task you'd like to do: 1

Enter the name of the customer: Beth Harmon
Enter customer's phone no: 1234568978
-------------------------------------------------------------------------------------------------------
S.N.  Laptop Name          Company Name         Price      Quantity   Graphics            CPU
-------------------------------------------------------------------------------------------------------
1     Razer Blade          Razer                $2000      75         i7 7th Gen          GTX 3060
2     XPS                  Dell                 $1976      26         i5 9th Gen          GTX 3070
3     Alienware            Alienware            $1978      49         i5 9th Gen          GTX 3070
4     Swift 7              Acer                 $900       67         i5 9th Gen          GTX 3070
5     Macbook Pro 16       Apple                $3500      80         i5 9th Gen          GTX 3070
-------------------------------------------------------------------------------------------------------

Enter the S.N. of the laptop that customer wants to buy: 1
Enter the quantity that customer wants to buy: 10
Laptop stock updated.

Does Customer want to purchase another laptop? (y/n): y
-------------------------------------------------------------------------------------------------------
S.N.  Laptop Name          Company Name         Price      Quantity   Graphics            CPU
-------------------------------------------------------------------------------------------------------
1     Razer Blade          Razer                $2000      65         i7 7th Gen          GTX 3060
2     XPS                  Dell                 $1976      26         i5 9th Gen          GTX 3070
3     Alienware            Alienware            $1978      49         i5 9th Gen          GTX 3070
4     Swift 7              Acer                 $900       67         i5 9th Gen          GTX 3070
5     Macbook Pro 16       Apple                $3500      80         i5 9th Gen          GTX 3070
-------------------------------------------------------------------------------------------------------
```

Figure 24: Laptop update after sales.

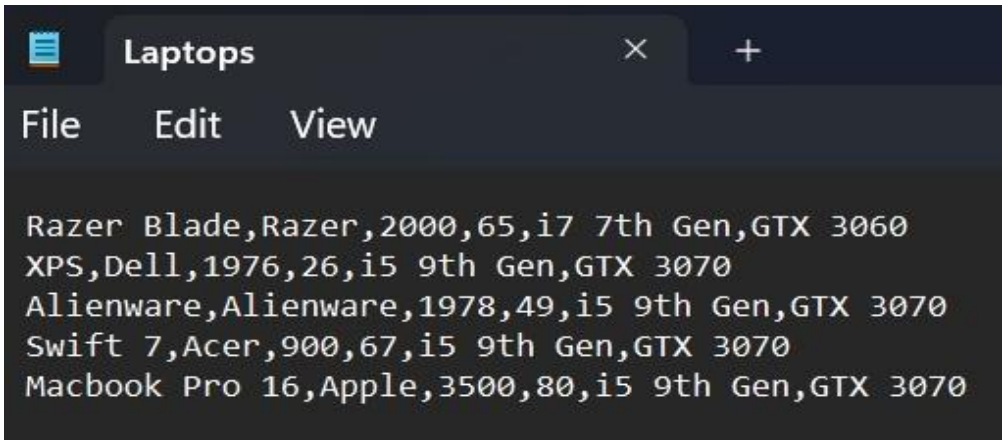

Figure 25: Updated product txt file.

Objective	To show the quantity being deducted while selling the laptop with updated laptops.txt
Action	Entering the task which perform the sale
Expected Result	The quantity should be deducted when selling laptop and should update laptops.txt also
Actual Result	The quantity was deducted when selling laptop and laptops.txt was updated
Conclusion	Successful

Table 5: Test 5 (sale update)

- Purchase

```
1. Customer's sale service.
2. Order laptops from the manufacture.
3. Exit the store.

You're in control. Select a number to indicate the task you'd like to do: 2

Enter your name admin: Karen white
-------------------------------------------------------------------------------------------------
S.N.  Laptop Name          Company Name         Price      Quantity   Graphics            CPU
-------------------------------------------------------------------------------------------------
1     Razer Blade          Razer                $2000      64         i7 7th Gen          GTX 3060
2     XPS                  Dell                 $1976      26         i5 9th Gen          GTX 3070
3     Alienware            Alienware            $1978      49         i5 9th Gen          GTX 3070
4     Swift 7              Acer                 $900       67         i5 9th Gen          GTX 3070
5     Macbook Pro 16       Apple                $3500      80         i5 9th Gen          GTX 3070
-------------------------------------------------------------------------------------------------

Enter the S.N. of the laptop you want to purchase from the manufacturer: 2
Enter the quantity you want to purchase: 10
Laptop stock updated.

Admin, do you want to order another laptop? (y/n): y
-------------------------------------------------------------------------------------------------
S.N.  Laptop Name          Company Name         Price      Quantity   Graphics            CPU
-------------------------------------------------------------------------------------------------
1     Razer Blade          Razer                $2000      64         i7 7th Gen          GTX 3060
2     XPS                  Dell                 $1976      36         i5 9th Gen          GTX 3070
3     Alienware            Alienware            $1978      49         i5 9th Gen          GTX 3070
4     Swift 7              Acer                 $900       67         i5 9th Gen          GTX 3070
5     Macbook Pro 16       Apple                $3500      80         i5 9th Gen          GTX 3070
-------------------------------------------------------------------------------------------------
```

Figure 26: Laptop update after purchase.

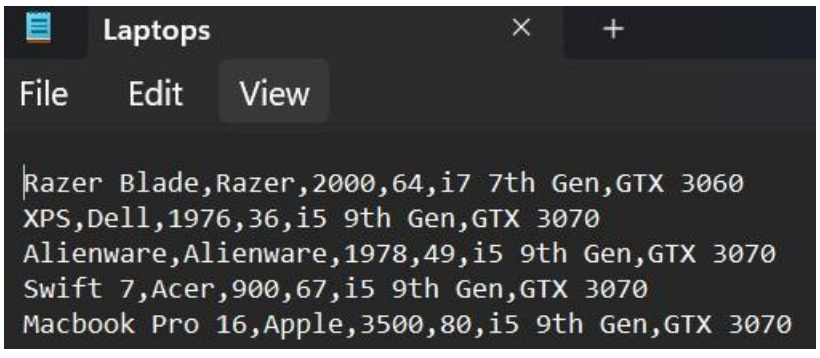

Figure 27: Updated product txt file.

Objective	To show the quantity being added while purchasing the laptop with updated laptops.txt
Action	Entering the task which perform the purchase
Expected Result	The quantity should be added when purchasing laptop and should update laptops.txt also
Actual Result	The quantity was added when selling laptop and laptops.txt was updated
Conclusion	Successful

Table 6: Test 5 (purchase update)

5. Conclusion:

In summary, the laptop shop requires a software program to effectively manage its inventory and transactions. The program should read and update a text file with laptop information and handle sales and purchase transactions. For sales, it should generate a detailed note with information such as the laptop’s name, brand, price, quantity, processor, graphics card, customer details, and the total amount including shipping costs. For purchases, the program should produce a note including the distributor's name, laptop details, purchase date, net amount, VAT, and gross amount. This software will ensure accurate and efficient management of the shop's inventory and transactions.

6. Pseudocode:

- Main.py

```
IMPORT datetime, read, write, operation

SET time to current date and time

SET laptop_dict to the dictionary returned by ReadingLaptopText function

CALL shop_header function from operation module

SET loop to True

WHILE loop is True
    PRINT "1. Customer's sale service."
    PRINT "2. Order laptops from the manufacturer."
    PRINT "3. Exit the store. \n"
    WHILE True
    TRY to convert user input to integer and assign to task
    IF error occurs
        PRINT error message
    END IF
    ELSE
        BREAK out of the loop
END WHILE
IF task is 1
    GET customer's name and phone number from user input
    WHILE name is empty
```

PRINT "Name cannot be empty."

GET customer's name from user input

END WHILE

WHILE length of phone number is not 10 or phone number contains non-digit characters

PRINT "Phone number must be a 10-digit number."

GET customer's phone number from user input

CONVERT phone number to integer

END WHILE

SET boughtlaptops to empty list

SET continueValue to True

WHILE continueValue is True

CALL available_laptops function from read module to display available laptops

CALL get_laptop_details function from operation module to get laptop details

GET laptop data, price, quantity, current quantity and laptop choice from get_laptop_details function

SET BrandName to the brand name of the laptop

SET laptopName to the name of the laptop

SET userQuantity to the quantity of laptops the customer wants to buy

SET unitPrice to the price of one laptop

SET totalPrice to the total price of the laptops the customer wants to buy

APPEND (BrandName, laptopName, userQuantity, unitPrice, totalPrice) tuple to boughtlaptops list

SET invoice_sale to the invoice file name with current date and time

UPDATE the quantity of the selected laptop in laptop_dict

```
PRINT "Laptop stock updated."
CALL WritingUpdatedtxt function from write module to write the updated laptop_dict to file
WHILE True
    GET answer from user input
    IF answer is "n"
        CALL MakingSaleInvoice function from write module to make sale invoice
        CALL Sale_invoice_details function from read module to display sale invoice details
        SET continueValue to False
        BREAK out of the loop
    END IF
    ELSE IF answer is "y"
        BREAK out of the loop
        END ELSE IF
        ELSE
            PRINT "Invalid option!"
        END WHILE
    END IF
IF task is 2
    GET admin's name from user input
    WHILE name is empty
        PRINT "Name cannot be empty."
        GET admin's name from user input
```

END WHILE

SET laptopsOrder to empty list

SET continueValue to True

WHILE continueValue is True

    CALL available_laptops function from read module to display available laptops

    CALL get_laptop_details function from operation module to get laptop details

    GET laptop data, price, quantity, current quantity and laptop choice from get_laptop_details function

    SET BrandName to the brand name of the laptop

    SET laptopName to the name of the laptop

    SET userQuantity to the quantity of laptops the admin wants to order

    SET unitPrice to the price of one laptop

    SET totalPrice to the total price of the laptops the admin wants to order

    APPEND (BrandName, laptopName, userQuantity, unitPrice, totalPrice) tuple to laptopsOrder list

    SET invoice_purchase to the invoice file name with current date and time

    UPDATE the quantity of the selected laptop in laptop_dict

    PRINT "Laptop stock updated."

    CALL WritingUpdatedtxt function from write module to write the updated laptop_dict to file

    WHILE True

        GET answer from user input

        IF answer is "n"

            CALL MakingPurchaseInvoice function from write module to make purchase invoice

            CALL Purchase_invoice_details function from read module to display purchase invoice details

```
            SET continueValue to False
            BREAK out of the loop
        END IF
        ELSE IF answer is "y"
            BREAK out of the loop
        END ELSE IF
        ELSE
            PRINT "Invalid option!"
    END WHILE
END IF
```

- Write.py:

```
Define MakingPurchaseInvoice in which name, laptopsOrder, time, invoice_purchase has been passed
  Open invoice_purchase and write it, invoice_purchase is defined as file
   SET total_amount to 0
   FOR each laptop in laptopsOrder
      CONVERT the laptop's unit price to an integer and add it to total_amount
   END FOR
   SET vat_rate to 0.13
   SET vat_amount to the product of total_amount and vat_rate
   SET gross_amount to the sum of total_amount and vat_amount
   CALL invoice_header function from operation module and write it to the file
   WRITE "Admin Name: " followed by the name to the file
```

WRITE "Ordered time: " followed by the formatted time to the file

WRITE a line break to the file

WRITE "-" 90 times to the file

SET headers to a tuple containing "Laptop Brand", "Item Name", "Total Quantity", "Unit Price", "Total"

WRITE headers to the invoice file using formatted string

WRITE "-" 90 times to the file

FOR each laptop in laptopsOrder

    SET details to a tuple containing the laptop's brand name, name, quantity, unit price, and total price

    WRITE the details to the invoice file using formatted string

END FOR

WRITE "-" 90 times to the file

    WRITE a line break to the file

    WRITE "Total amount: $" followed by total_amount to the file

    WRITE "VAT amount (13%): $" followed by vat_amount to the file

    WRITE "Gross Amount: $" followed by gross_amount to the file

    WRITE "Admin " followed by the name and " authorized this transaction." to the file

    WRITE "-" 90 times to the file

END FUNCTION

Define MakingSaleInvoice in which name, phone_num, boughtlaptops, time, invoice_sale has been passed

    Open invoice_sale and write it, invoice_sale is defined as file

    SET total_amount to 0

SET shippingCost to 100

FOR each laptop in boughtlaptops

    CONVERT the laptop's unit price to an integer and add it to total_amount

END FOR

SET grandTotal to the sum of total_amount and shippingCost

CALL invoice_header function from operation module and write it to the file

WRITE "Customer Name: " followed by the name to the file

WRITE "Phone no: " followed by the phone_num to the file

WRITE "Sold time: " followed by the formatted time to the file

WRITE "-" 90 times to the file

SET headers to a tuple containing "Laptop Brand", "Item Name", "Total Quantity", "Unit Price", "Total"

WRITE headers to the invoice file using formatted string

WRITE "-" 90 times to the file

FOR each laptop in boughtlaptops

    SET details to a tuple containing the laptop's brand name, name, quantity, unit price, and total price

    WRITE the details to the invoice file using formatted string

END FOR

WRITE "-" 90 times to the file

WRITE a line break to the file

WRITE "Shipping cost: $" followed by shippingCost to the file

WRITE "Grand Total: $" followed by grandTotal to the file

WRITE "-" 90 times to the file

```
    IF laptop_dict is not empty:
        OPEN "Laptops.txt" in write mode as file
        FOR laptop_info in laptop_dict.values() DO:
            CONCATENATE the values in laptop_info list with a comma separator and a newline character
            WRITE the concatenated string to file
        END FOR
        CLOSE file
END FUNCTION
```

- read.py:

```
FUNCTION ReadingLaptopText()
    OPEN "Laptops.txt" file for reading as file
    SET laptop_dict to an empty dictionary
    SET num to 1
    FOR each line in file
        REMOVE the newline character from the end of the line
        SPLIT the line by comma into a list of values
        ADD the list of values to the laptop_dict dictionary with key num
        INCREMENT num by 1
    END FOR
    CLOSE the file
    RETURN laptop_dict
END FUNCTION


FUNCTION Purchase_invoice_details(invoice_purchase)
    OPEN invoice_purchase file for reading as file
    READ the contents of the file into a variable named contents
    PRINT contents
    CLOSE the file
END FUNCTION
FUNCTION Sale_invoice_details(invoice_sale)
```

```
    OPEN invoice_sale file for reading as file
    READ the contents of the file into a variable named contents
    PRINT contents
    CLOSE the file
END FUNCTION

FUNCTION available_laptops()
    PRINT a header row for the laptop table
    OPEN "Laptops.txt" file for reading as file
    SET n to 0
    FOR each line in file
        SPLIT the line by comma into a list of values
        SET price to the third value of the list
        ADD a dollar sign to the beginning of price
        REPLACE the third value of the list with price
        INCREMENT n by 1
        PRINT a row of laptop data using the values in the list
    END FOR
    CLOSE the file
    PRINT a separator line for the laptop table
END FUNCTION
```

- operation.py:

```
FUNCTION get_laptop_details(laptop_dict)
    WHILE True:
        TRY:
            PROMPT user for input with "Enter the S.N. of the laptop that customer wants to buy: "
            CONVERT input to integer and store in laptop_choice
            IF laptop_choice not in laptop_dict THEN
                PRINT "Invalid S.N. Please enter a valid S.N."
                CONTINUE to the next iteration of the loop
        EXCEPT ValueError:
            PRINT "We're sorry, but the S.N you entered is not valid. Please enter a numeric value."
            CONTINUE to the next iteration of the loop
    SET laptop_data to the value associated with the laptop_choice key in laptop_dict
    SET current_quantity to the integer value of the fourth element in laptop_data
    WHILE True:
        TRY:
            PROMPT user for input with "Enter the quantity that customer wants to buy: "
            CONVERT input to integer and store in quantity
            IF quantity <= 0 or quantity > current_quantity THEN
                PRINT "Sorry, the requested quantity is not available in our shop."
            ELSE:
                BREAK out of the inner loop
```

```
            EXCEPT ValueError:
                PRINT "We're sorry, but the quantity you entered is not valid. Please enter a numeric value."
        END WHILE
        SET price to the value of the third element in laptop_data
        BREAK out of the outer loop
    END WHILE
    RETURN laptop_data, price, quantity, current_quantity, laptop_choice
END FUNCTION
FUNCTION get_laptop_detail(laptop_dict)
    WHILE True:
        TRY:
    PROMPT user for input with "Enter the S.N. of the laptop you want to purchase from the manufacturer: "
    CONVERT input to integer and store in laptop_choice
    IF laptop_choice not in laptop_dict THEN
        PRINT "Invalid S.N. Please enter a valid S.N."
        CONTINUE to the next iteration of the loop
EXCEPT ValueError:
    PRINT "We're sorry, but the S.N you entered is not valid. Please enter a numeric value."
    CONTINUE to the next iteration of the loop
SET laptop_data to the value associated with the laptop_choice key in laptop_dict
WHILE True:
    TRY:
```

```
        PROMPT user for input with "Enter the quantity you want to purchase: "
        CONVERT input to integer and store in quantity
        IF quantity <= 0 THEN
            PRINT "Sorry admin!. The quantity is invalid"
        ELSE:
            SET current_quantity to the integer value of the fourth element in laptop_data
            BREAK out of the inner loop
    EXCEPT ValueError:
        PRINT "We're sorry, but the quantity you entered is not valid. Please enter a numeric value."
END WHILE
SET price to the value of the third element in laptop_data
        BREAK out of the outer loop
    END WHILE
    RETURN laptop_data, price, quantity, current_quantity, laptop_choice
END FUNCTION
SET company_name to "Pacific Technology"
SET location to "Ocean Boulevard Street, South Carolina, USA"
SET phone_number to "555-1234"
SET email to "PacificTech1995@gmail.com"
FUNCTION shop_header()
    PRINT "*" repeated 210 times
    PRINT " " repeated 90 times, followed by company_name
```

```
    PRINT " " repeated 80 times, followed by location

    PRINT " " repeated 77 times, followed by "Phone: ", phone_number, " Email: ", email

    PRINT "*" repeated 210 times

    PRINT a blank line

    PRINT "Welcome back, admin! Your expertise and attention to detail keep our PC store running smoothly."

END FUNCTION

FUNCTION invoice_header(file)

    PRINT "*" repeated 90 times

    PRINT " " repeated 40 times, followed by company_name

    PRINT " " repeated 30 times, followed by location

    PRINT " " repeated 27 times, followed by "Phone: ", phone_number, " Email: ", email

    PRINT "*" repeated 90 times

    PRINT a blank line

END FUNCTION
```