EPC_E8.1

Mobile Phone Store Management Program

1. Data Structure [2.5 mark]

+ Create a data structure with the following specification:

struct **Mobile**		
Data Type	Information Field	Description
char	**name[25]**	Mobile’s name
char	**manufacturer[25]**	Manufacturer
int	**release_year**	Release year
float	**price**	Mobile’s price

2. Menu-based Program [2.5 mark]

+ Create a menu based program as follows:

```
+-------------------------------------------------------------------+
|                  MOBILE PHONE STORE MANAGEMENT PROGRAM            |
+-------------------------------------------------------------------+
|1. Input |2. Sort |3. Analyze |4. Find |5. Save |6. Open |7. Exit |
+-------------------------------------------------------------------+
```

Note:

+The menu is displayed when the program starts.

+After each choice of the user is completed, ask him/her:

```
Do you want to continue ?
- Yes, I do. (press ‘y’, ‘Y’)
- No, I don’t. (press ‘n’, ‘N’)
- Please clear the screen ! (press ‘c’, ‘C’)
Your choice:
```

+If the user enters:

- ‘y’ or ‘Y’: repeat the menu.

-‘n’,‘N’, or ‘7’: exit the program.

-‘c’ or ‘C’: clear the screen.

+Ask the user to re-enter when his/her choice is invalid.

3. Input a list of Mobiles [2.5 mark]

When user selects the option “**1. Input”** from the menu, do the followings:

- Ask the user to enter the total number of **Mobiles.**
- Input information for each **`Mobile`** in the list.
- Validate data input by the following rules:

Data Anomalies	Message
release_year < 1999	Mobile release year must be greater than or equal to 1999.
price < 500 or price > 6000	Mobile price must be between 500($) and 6000($).

- If the user enters invalid data, ask him/her to re-enter.

Note: This module must be coded in a function whose parameter is a pointer to the `struct` **`Mobile.`**

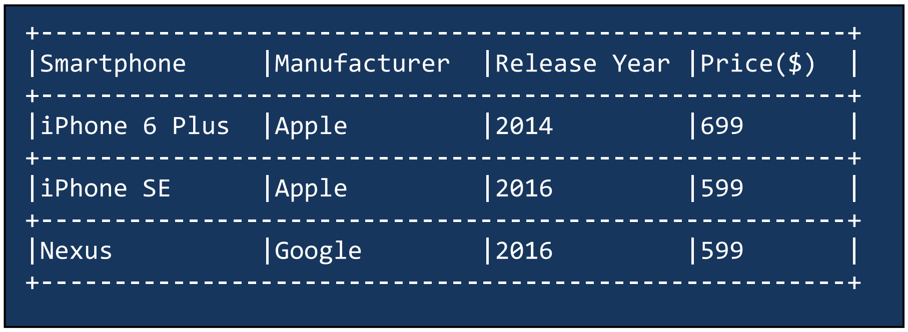
Input Example: (texts in **bold** are input by user at runtime)

```
Please input Mobile[1]:
Name: iPhone 6 Plus
Manufacturer: Apple
Release Year: 2014
Price: 699
```

4. Sort Mobiles by name in ascending order [2.5 mark]

When user selects the option “**2. Sort**” from the the menu:

- Sort Mobiles by name in ascending order (from A-Z).
- Displays details of each Mobile in the list after the above re-arrangement. For example:



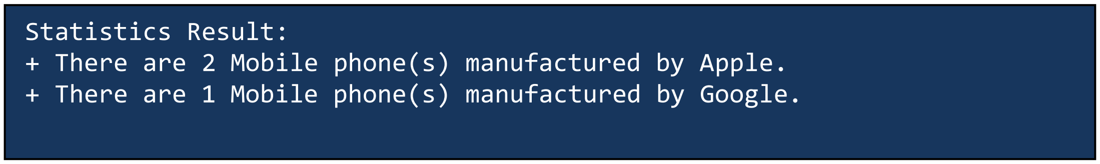
```
+-----------------------------------------------------+
|Smartphone     |Manufacturer  |Release Year |Price($)  |
+-----------------------------------------------------+
|iPhone 6 Plus  |Apple         |2014         |699       |
+-----------------------------------------------------+
|iPhone SE      |Apple         |2016         |599       |
+-----------------------------------------------------+
|Nexus          |Google        |2016         |599       |
+-----------------------------------------------------+
```

Note: This module must be coded in a function whose parameter is a pointer to the `struct` **`Mobile.`**

5. Analyze statistics of Mobiles by manufacturer [2.5 mark]

When user selects the option “**3. Analyze”** from the menu :

- Calculate how many Mobiles are there from a specific manufacturer.
- Display the statistics results. For example:



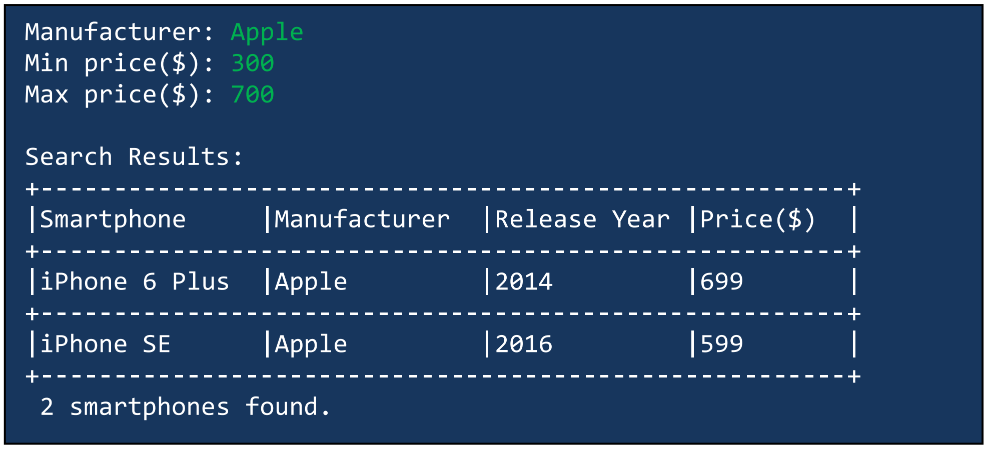
```
Statistics Result:
+ There are 2 Mobile phone(s) manufactured by Apple.
+ There are 1 Mobile phone(s) manufactured by Google.
```

Note: This module must be coded in a function whose parameter is a pointer to the `struct` **`Mobile.`**

6. Find Mobiles by manufacturer and price [2.5 mark]

When user selects the option “**4. Find”** from menu:

- Ask the user to input a *manufacturer*, a *min* and a *max* price.
- Display all the Mobiles that are from the *manufacturer* (input by user) with prices between *min* and *max*. For example:



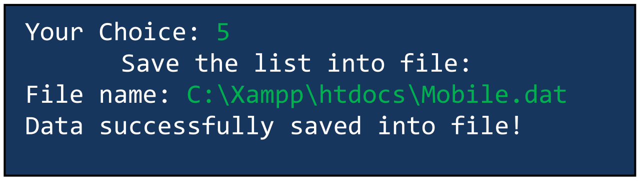
```
Manufacturer: Apple
Min price($): 300
Max price($): 700

Search Results:
+--------------------------------------------------------+
|Smartphone     |Manufacturer   |Release Year |Price($)  |
+--------------------------------------------------------+
|iPhone 6 Plus  |Apple          |2014         |699       |
+--------------------------------------------------------+
|iPhone SE      |Apple          |2016         |599       |
+--------------------------------------------------------+
 2 smartphones found.
```

Note: This module must be coded in a function whose parameter is a pointer to the `struct` **`Mobile.`**

7. Save to file [2.5 mark]

When user selects the option **“5. Save”** from the menu, do the followings:

- Ask the user to enter a file name.
- Create the file in binary mode.
- If the program cannot open file, notify the user about that error and return to end this function.
- Write data of the list into the file.
- Notify the user that data has been saved successfully.
- Close the file.



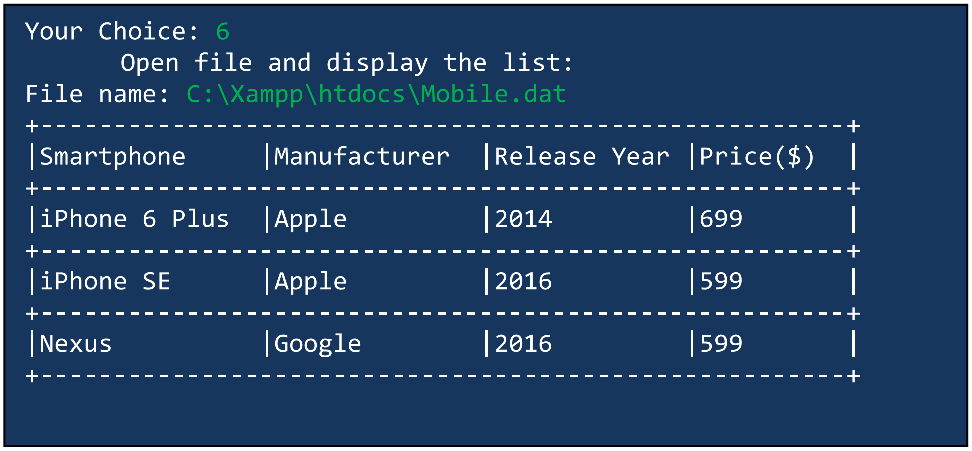
```
Your Choice: 5
      Save the list into file:
File name: C:\Xampp\htdocs\Mobile.dat
Data successfully saved into file!
```

Note: This module must be coded in a function whose parameter is a pointer to the `struct` **`Mobile`**.

8. Open File and display the list [2.5 mark]

When user selects the option **"6. Open"** from the menu, do the followings:

- Ask the user to enter a file name.
- Open the file in binary mode.
- If the program cannot open file, notify the user about that error and return to end this function.
- Read data out of the file, fill into the list and display them all.
- Notify the user that data has been saved successfully.
- Close the file.



```
Your Choice: 6
      Open file and display the list:
File name: C:\Xampp\htdocs\Mobile.dat
+-------------------------------------------------------+
|Smartphone     |Manufacturer  |Release Year |Price($)  |
+-------------------------------------------------------+
|iPhone 6 Plus  |Apple         |2014         |699       |
+-------------------------------------------------------+
|iPhone SE      |Apple         |2016         |599       |
+-------------------------------------------------------+
|Nexus          |Google        |2016         |599       |
+-------------------------------------------------------+
```

Note: **The user must input the list of Mobiles before selecting any other options from the menu, except** "`Clear screen`" **and** "`Exit`"**.**

Vocabulary

declare	Khai báo	**parameter**	Đối số, tham số
follows , following	Như sau	**structure array**	Mảng cấu trúc
arrange	Sắp xếp	**structure pointer**	Con trỏ cấu trúc
manage	Quản lý	**store, save**	Lưu trữ, lưu
Mobile	Điện thoại thông minh	**format**	Định dạng

manufacturer	Nhà sản xuất	**module**	Phần, câu, mô đun
price	giá	**information**	Thông tin
release year	Năm phát hành	**include**	Bao gồm
calculate	Tính toán	**create**	Tạo
ascending order	Thứ tự tăng dần	**include**	Bao gồm
fail	Thất bại	**analyze statistics**	Phân tích thống kê
validate	Xác thực	**invalid**	Không hợp lệ
notify	Thông báo		