CSC128 – FUNDAMENTALS OF COMPUTER PROBLEM SOLVING

CREATIVE PHONECASE

LECTURER'S NAME: MADAM RASHIDAH BINTI MOKHTAR

CLASS: JSC1101I

NAME	MATRIX NUM.
NURAIN NABILAH BINTI SALEHUDDIN	2018258486
SITI HAJAR BINTI KAMARUZAMAN	2018271502
SITI NORHAZLINA BINTI MOHD HALIL	2018205256
NUR ZAFIRAH SYAMIMI BINTI NORMAN	2018402326

TABLE OF CONTENTS

1.0 – INTRODUCTION

1.1 Project Background

First and foremost we would like to appreciate and thanked to our beloved Lecturer in charge for our class, Madam Rashidah Binti Mokhtar for guiding and leading us on this whole project for our CSC128 which is Fundamental of Computer Problem Solving. This group consist of four members who are Nurain Nabilah Bt Salehuddin, Siti Hajar Bt Kamaruzaman, Nur Zafirah Syamimi Bt Norman and last but not least Siti Norhazlina Bt Mohd Halil. This project proposal report started 15 September 2018.

The title of our project is *"CREATIVE PHONECASE*" where we provide designing and printing service for the phone casing. This program lists multiple choices of types of design (customize or original design) types of phone model (XIAOMI, OPPO, VIVO, SAMSUNG, LENOVO, HUAWEI, IPHONE), type of printing (screen printing, embossing printing, debossing printing and 3D sublimation) and types of material (Plastic, Thermo-polyurathane, and carbon fiber).

There will also be input and output for shipping activities such as; statements of personal information such as name, IC numbers, address, phone number, email address.

This program also include a system that will calculate total price of the item purchased by customers, no matter how many phonecases they buy. The price is according to the type of designs, model of the phone, material of the phonecases and type of printing. It will also calculate the selling price after discounts if they are a member of the store. This system will also calculate the sum of the whole price including the charges (delivery + the store's own charges) with a fixed price which is RM10. At the end of purchasing process the customers will see the total price of the items they bought and other informations related to their purchases.

1.2 Objectives

- To calculate the total of items sold in a day.
- To display the informations of the customers for delivery purposes.
- To calculate the total price of all items bought by the customers.
- To calculate the selling price of item including charges on the delivery.
- To calculate the deduction or price after discount for every event.

1.3 Scope & Program content

SCOPE:

Our targeted users are most millennials generation as this generation love following new trends and always open to new creative world. They are always the ones who are interested in showing off their identity using their phonecase designs it is also one of a way to show their interest in something.

As a qoute taken from a website as a reference “This trend is about life not just fashion.” And the escalating importance of how our phones look reflects the fact that the phone is as much a part of our visual world as our functional one.

For the users of this program, they would find it is more easier and efficient to find design that are suitable to their likings or they can even design according to their own creative ideas.

PROGRAM CONTENT:

CONTENT	DESCRIPTION	FURTHER DESCRIPTION
Items	Phonecases	-
Classification	Types of Phone Model	XIAOMI,OPPO,VIVO,SAMSUNG,LENOVO, IPHONE.
	Types of Material	Plastic,Thermo-polyurathane,carbon fiber
	Types of Printing	Screen printing, embossing printing, debossing printing and 3D sublimation.
	Types of Designs	Customize or original designs
Calculation	Price	Affordabble for customers
	Discounts	On special events
	Charges	Delivery and the store’s own charges
	Sales	Total sales for a day
	Counting	• Total customers • Total phonecases sold

TABLE ON THE PRICING FOR THE ITEMS:

Types of Designs	Phone's Model	Type of Printing	Case's Material	Original Price	Price WitH Charges (+RM10)
Customize or *Original*	*SAMSUNG* or *OPPO* or *VIVO* or *XIAOMI* or *IPHONE* or *HUAWEI* or *LENOVO*	Screen printing	Plastic	RM10	RM20
			Thermo-polyurathane	RM20	RM30
			Carbon fiber	RM15	RM25
		Embossing printing	Plastic	RM15	RM25
			Thermo-polyurathane	RM23	R M23
			Carbon fiber	RM18	RM28
		Debossing printing	Plastic	RM15	RM25
			Thermo-polyurathane	RM23	RM33
			Carbon fiber	RM18	RM28
		3D sublimation	Plastic	RM20	RM30
			Thermo-polyurathane	RM26	RM36
			Carbon fiber	RM23	RM33

TABLE ON THE CALCULATION:

Classification	Types of Calculation	Formula	Formula (Coding)
Price with charges	-Summation	sellingPrice = price + RM10	sellingPrice = price + 10.00;

Total Price (no discount)	-Summation	totPrice = 0
		totPrice = totPrice + sellingPrice
			totPrice = 0;
			totPrice = totPrice + sellingPrice;

Discount	-Multiplication	Discount = 20% x totPrice	Discount = 0.20 * totoPrice;
Price with discount	-Subtraction	totalPrice= totPrice – Discount	totalPrice = totPrice - Discount

Total Sales	-Summation	totSales = 0
		totSales = totSales + totalPrice
			totSales = 0;
			totSales = totSales + totalPrice;

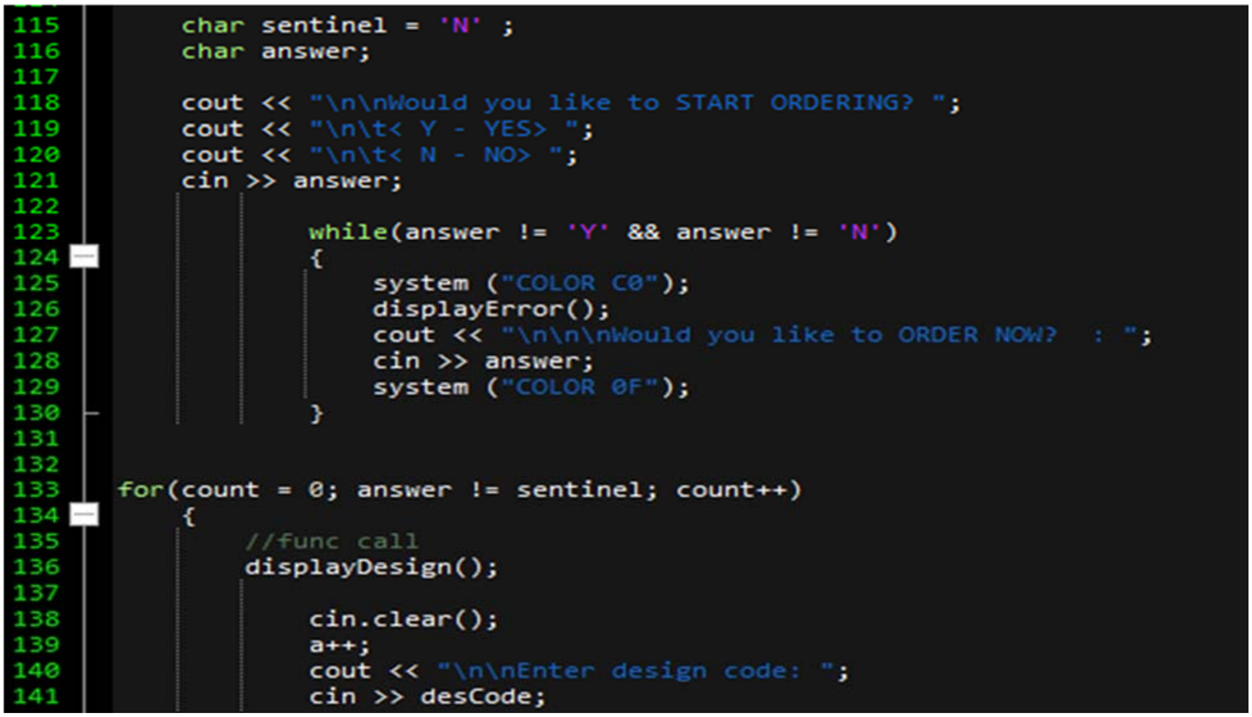
2.0 – IMPLEMENTATION

2.1 – LIST OF REPETITIONS

- This program use **loop control stuctures** which include;
    - **sentinel** controlled loop
    - **counter** controlled loop

- *Sentinel-Controlled Looping*

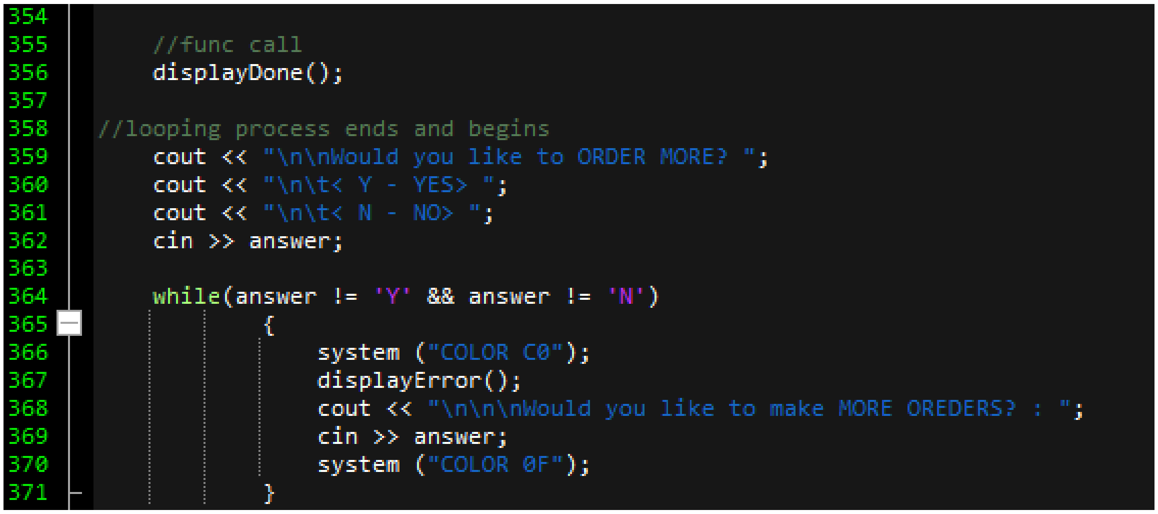
1. The first sentinel controlled loop is used for the *inner loop* which is for the looping of the next item to be sold.



```
    char sentinel = 'N' ;
    char answer;

    cout << "\n\nWould you like to START ORDERING? ";
    cout << "\n\t< Y - YES> ";
    cout << "\n\t< N - NO> ";
    cin >> answer;

            while(answer != 'Y' && answer != 'N')
            {
                system ("COLOR C0");
                displayError();
                cout << "\n\n\nWould you like to ORDER NOW?  : ";
                cin >> answer;
                system ("COLOR 0F");
            }


for(count = 0; answer != sentinel; count++)
    {
        //func call
        displayDesign();

            cin.clear();
            a++;
            cout << "\n\nEnter design code: ";
            cin >> desCode;
```

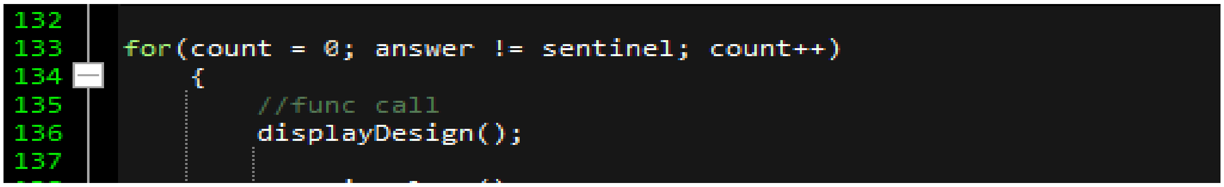
- Sentinel is set as ‘N’ which stands for “NO”. The loop used is `for` loop structure loop control variable (LCV) is being used to set the *sentinel*. The question asked if the user wants to start ordering , ‘Y’ or ‘N’ .

- If the user entered Y, the body of the loop will started to process.



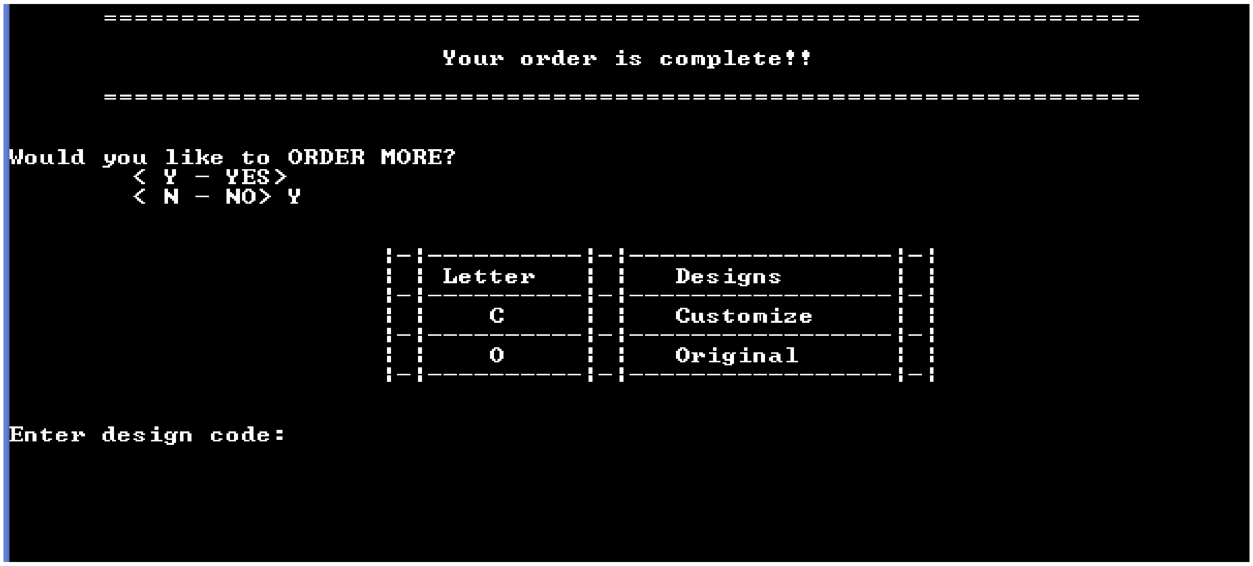
```
    //func call
    displayDone();

//looping process ends and begins
    cout << "\n\nWould you like to ORDER MORE? ";
    cout << "\n\t< Y - YES> ";
    cout << "\n\t< N - NO> ";
    cin >> answer;

    while(answer != 'Y' && answer != 'N')
            {
                system ("COLOR C0");
                displayError();
                cout << "\n\n\nWould you like to make MORE OREDERS? : ";
                cin >> answer;
                system ("COLOR 0F");
            }
```

- At the end of the body, another question was asked, “Would you like to order more?” . The cin variable used is the same as the first question which is answer.
- If you remember the loop condtion was;



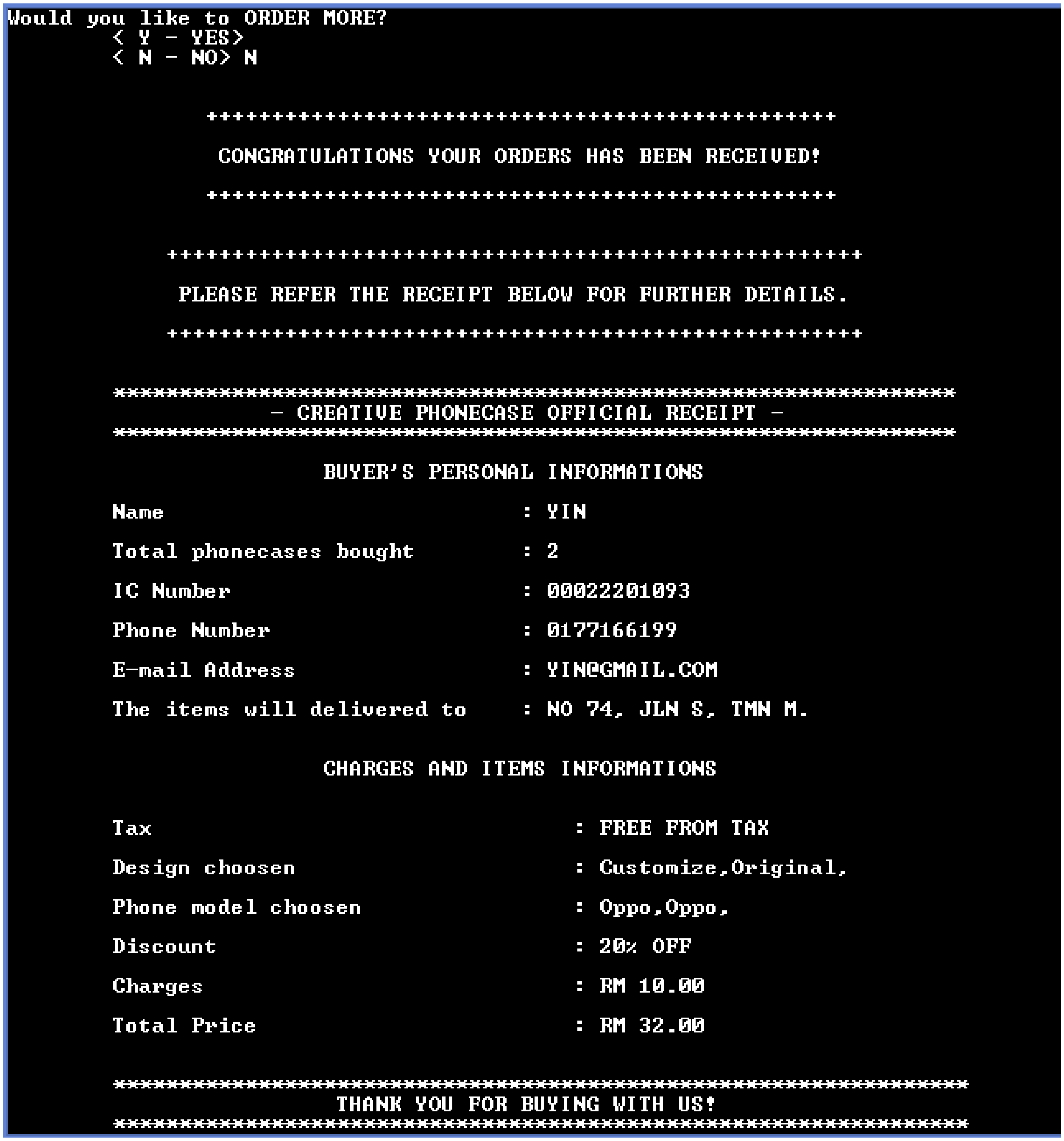
```
for(count = 0; answer != sentinel; count++)
    {
        //func call
        displayDesign();
```

- The LCV here is { answer != sentinel } for the loop to keep on repeating. Sentinel was set as ‘N’. This means if the answer was inputed as ‘Y’ . The body will loop once again which means the user can order the next item.



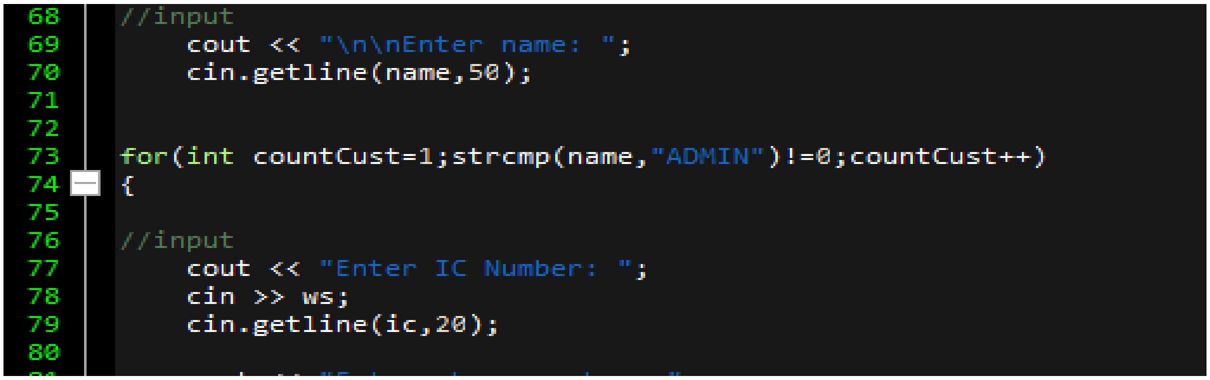
```
      ===================================================================

                            Your order is complete!!

      ===================================================================


Would you like to ORDER MORE?
        < Y - YES>
        < N - NO> Y

                        ¦-¦-----------¦-¦-----------------¦-¦
                        ¦ ¦ Letter    ¦ ¦    Designs      ¦ ¦
                        ¦-¦-----------¦-¦-----------------¦-¦
                        ¦ ¦     C     ¦ ¦    Customize    ¦ ¦
                        ¦-¦-----------¦-¦-----------------¦-¦
                        ¦ ¦     O     ¦ ¦    Original     ¦ ¦
                        ¦-¦-----------¦-¦-----------------¦-¦


Enter design code:
```

- As you can see, the first order was complete and the question was asked. The user *entered 'Y'* and the process *looped once again.*
- But, if the user *entered 'N'* as in the user entered the sentinel setted word. The loop will *stop* and will *display the receipt* to the user.

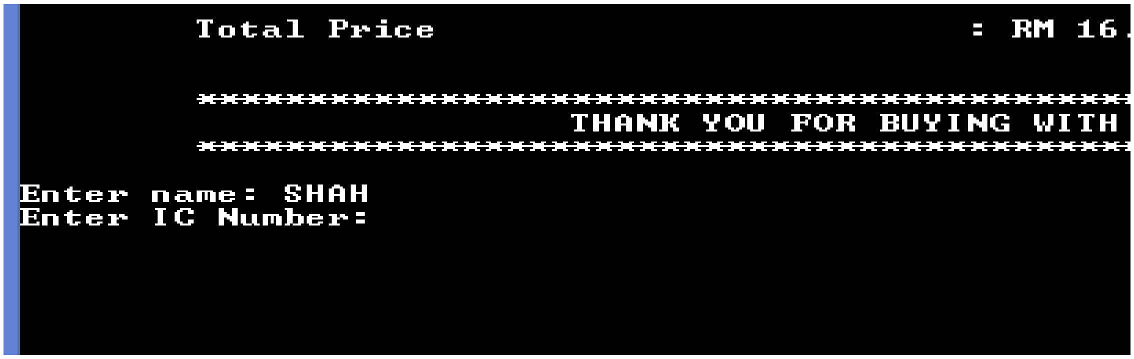


```
Would you like to ORDER MORE?
        < Y - YES>
        < N - NO> N


               ++++++++++++++++++++++++++++++++++++++++++++++++

                CONGRATULATIONS YOUR ORDERS HAS BEEN RECEIVED!

               ++++++++++++++++++++++++++++++++++++++++++++++++


            +++++++++++++++++++++++++++++++++++++++++++++++++++++

             PLEASE REFER THE RECEIPT BELOW FOR FURTHER DETAILS.

            +++++++++++++++++++++++++++++++++++++++++++++++++++++


        ****************************************************************
                    - CREATIVE PHONECASE OFFICIAL RECEIPT -
        ****************************************************************

                        BUYER'S PERSONAL INFORMATIONS

        Name                            : YIN

        Total phonecases bought         : 2

        IC Number                       : 00022201093

        Phone Number                    : 0177166199

        E-mail Address                  : YIN@GMAIL.COM

        The items will delivered to     : NO 74, JLN S, TMN M.

                        CHARGES AND ITEMS INFORMATIONS


        Tax                                 : FREE FROM TAX

        Design choosen                      : Customize,Original,

        Phone model choosen                 : Oppo,Oppo,

        Discount                            : 20% OFF

        Charges                             : RM 10.00

        Total Price                         : RM 32.00


        *****************************************************************
                         THANK YOU FOR BUYING WITH US!
        *****************************************************************
```

2. Next, the sentinel controlled loop that can be found in this program is in the *outer loop* for the next customer.

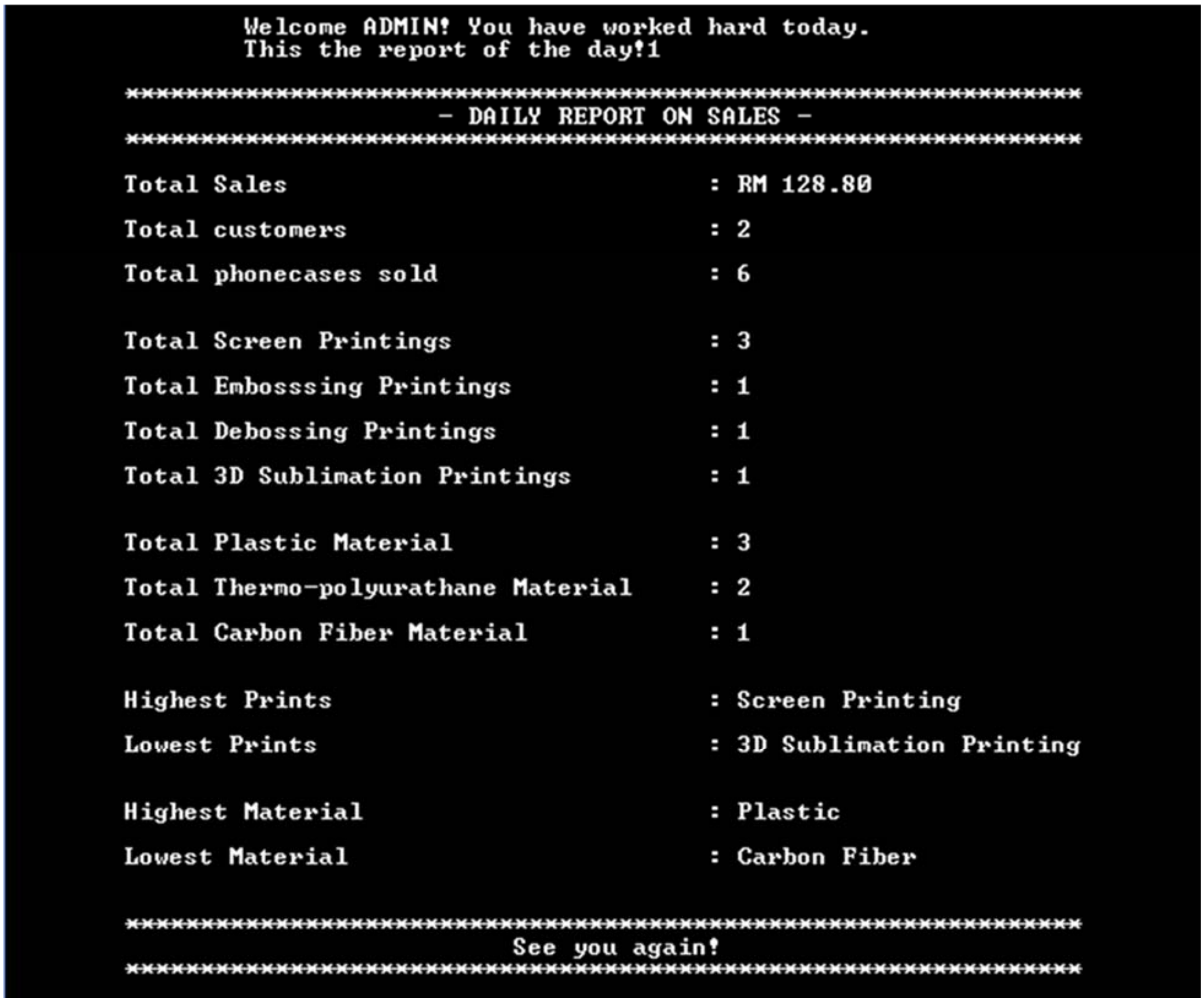


```
//input
    cout << "\n\nEnter name: ";
    cin.getline(name,50);


for(int countCust=1;strcmp(name,"ADMIN")!=0;countCust++)
{

//input
    cout << "Enter IC Number: ";
    cin >> ws;
    cin.getline(ic,20);

```

- As above, by using `for` loop structure the LCV is used as sentinel. The word “ADMIN” is actually the sentinel. But, we do not declare it in the program.
- The body of the loop statement will ***repeated*** as long as the “Name:” entered is ***other than “ADMIN”.***
- 

```
           Total Price                                     : RM 16.

           ************************************************************
                                 THANK YOU FOR BUYING WITH
           ************************************************************

Enter name: SHAH
Enter IC Number:
```

- As you can see, the first customer had finished the orders and got the receipt. The next *customer entered another name* which in this case “SHAH” . The body of the loop continue to run its process.

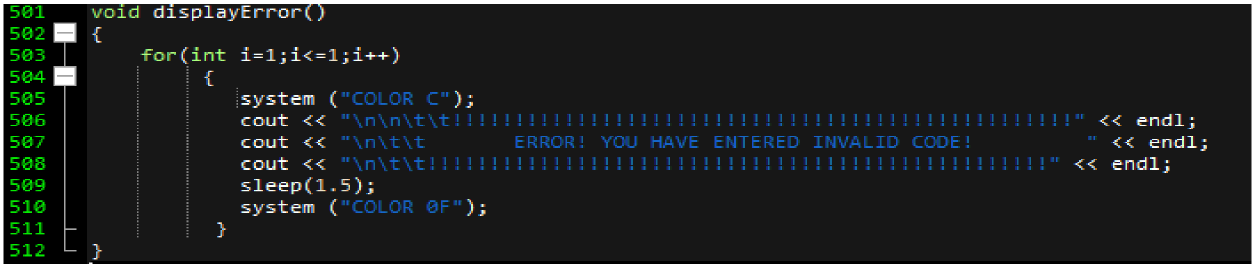


```
                Welcome ADMIN! You have worked hard today.
                This the report of the day!1

        ****************************************************************
                             - DAILY REPORT ON SALES -
        ****************************************************************

        Total Sales                            : RM 128.80

        Total customers                        : 2

        Total phonecases sold                  : 6


        Total Screen Printings                 : 3

        Total Embosssing Printings             : 1

        Total Debossing Printings              : 1

        Total 3D Sublimation Printings         : 1


        Total Plastic Material                 : 3

        Total Thermo-polyurathane Material     : 2

        Total Carbon Fiber Material            : 1


        Highest Prints                         : Screen Printing

        Lowest Prints                          : 3D Sublimation Printing


        Highest Material                       : Plastic

        Lowest Material                        : Carbon Fiber


        ****************************************************************
                                  See you again!
        ****************************************************************
```

- But, one the *word "ADMIN"* is entered the *loop will stop* and automatically will show an *output of daily report* just like in the snippet above.

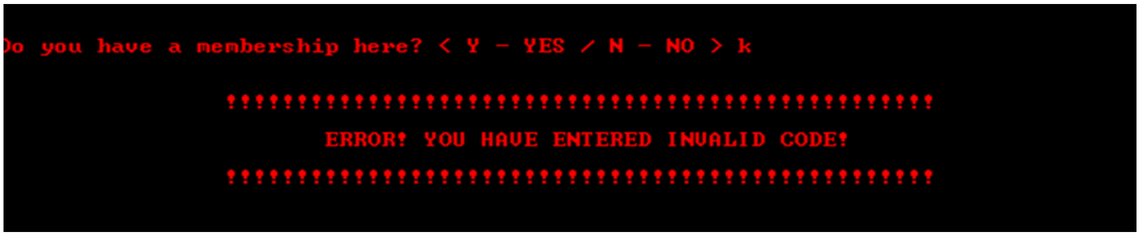
- *Counter-Controlled Loop*

1. One of the first counter controlled loop that was used is in the blinking **ERROR MESSAGE.**



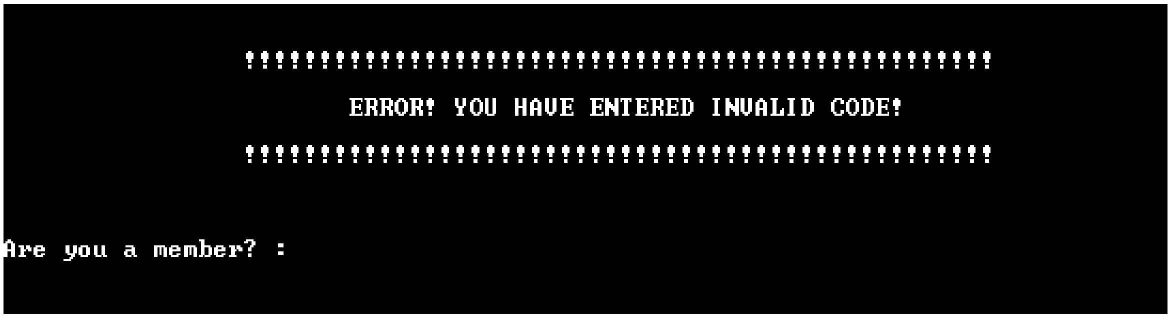
```
void displayError()
{
    for(int i=1;i<=1;i++)
        {
          system ("COLOR C");
          cout << "\n\n\t\t!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!" << endl;
          cout << "\n\t\t       ERROR! YOU HAVE ENTERED INVALID CODE!         " << endl;
          cout << "\n\t\t!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!" << endl;
          sleep(1.5);
          system ("COLOR 0F");
        }
}
```

- As you can see, once again the program used `for` loop structure where it initialized i as 1. Then the LCV was set as i<=1 for the loop to be valid condition to start and stop the repetition and i++ was set as increment of the LCV.
- At first, i was 1 . So, it was valid to loop. But after looping one time i++ made i became 2 which didn't pass the i must be less or equal to 1.
- The looping stops after it loops for 1 time.The main function here is to loop the color of the fonts. Below as shown below;



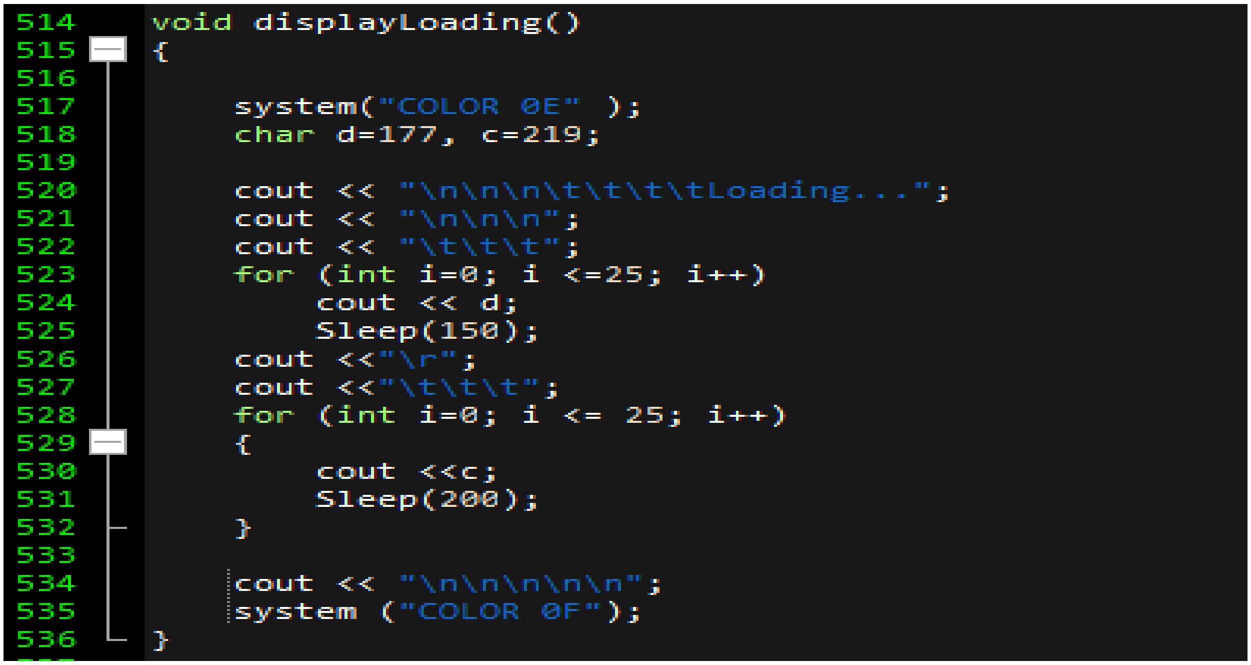
```
Do you have a membership here? < Y - YES / N - NO > k

                !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

                       ERROR! YOU HAVE ENTERED INVALID CODE!

                !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

- If the user entered other input besides 'Y' or 'N' , the error message will displayed. The font becomes red one time as it loops and sleep for 1.5 seconds.
- The message return to original color as the the loops stopped.



```
                !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

                       ERROR! YOU HAVE ENTERED INVALID CODE!

                !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!



Are you a member? :
```

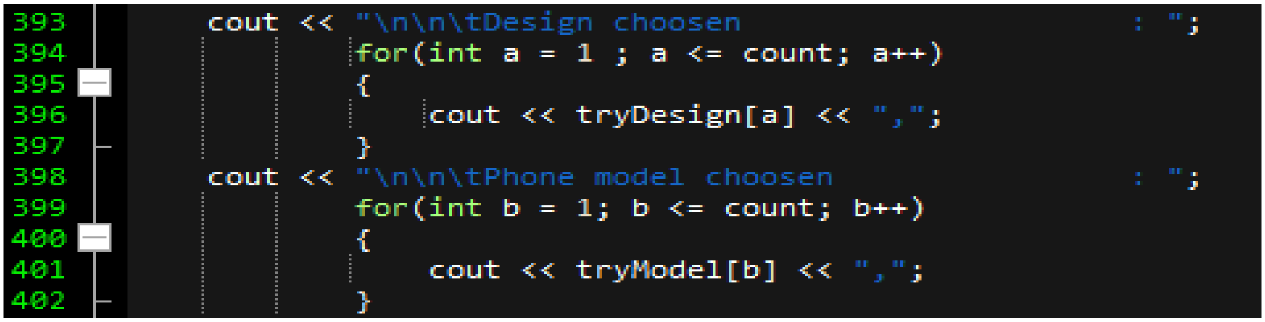
2. The next counter controlled loop is when the program uses **LOADING ARTWORK.**



```
void displayLoading()
{

    system("COLOR 0E" );
    char d=177, c=219;

    cout << "\n\n\n\t\t\t\tLoading...";
    cout << "\n\n\n";
    cout << "\t\t\t";
    for (int i=0; i <=25; i++)
        cout << d;
        Sleep(150);
    cout <<"\r";
    cout <<"\t\t\t";
    for (int i=0; i <= 25; i++)
    {
        cout <<c;
        Sleep(200);
    }

    cout << "\n\n\n\n\n";
    system ("COLOR 0F");
}
```

- The loop set i=0, the LCV was set as i less or equal 25 and i++ as an increment. The loop starts from 0 and then repeated 25 as the i++ increases and stops when reached the condition of the LCV.

3. The last counter controlled loop used was in the ARRAY.

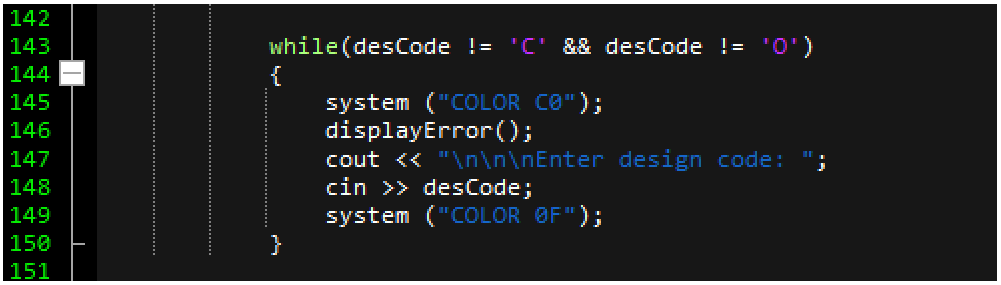


```
    cout << "\n\n\tDesign choosen                        : ";
            for(int a = 1 ; a <= count; a++)
            {
                cout << tryDesign[a] << ",";
            }
    cout << "\n\n\tPhone model choosen                   : ";
            for(int b = 1; b <= count; b++)
            {
                cout << tryModel[b] << ",";
            }
```

- In this array, a=1 or b=1. a and b is used as array index as well. So, in this case the index of the array starts at 1 not 0. Both of the LCV is set to less or equals to count and if the LCV was passed, a and b will be plus by 1 as the increment update.
- Count here represent the value of phonecases the user bought. The loop stops as the a or b reaches the same value as count.

- *Other types of looping*

1. *While...* loop was used to loop the display of **ERROR MESSAGES**.

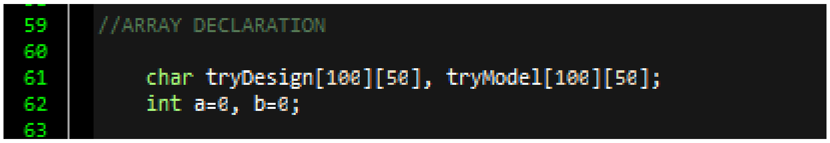


```
while(desCode != 'C' && desCode != 'O')
{
    system ("COLOR C0");
    displayError();
    cout << "\n\n\nEnter design code: ";
    cin >> desCode;
    system ("COLOR 0F");
}
```

- This structure are *used similarly* to *all classifications* in the program.
- After every input questions, this coding structure is used according to it's LCV.
- The example above is one of the coding. The LCV is set as desCode is not equal to 'C' and desCode not equal to 'O'.
- The *body will loop* as the user *entered other input besides 'C' and 'O'* . The user will see an *error message* and will asked *to re-enter the design code*. The body will continue to loop if the input stays wrong. It'll stop once the user entered 'C' or 'O' which is the right input.
-

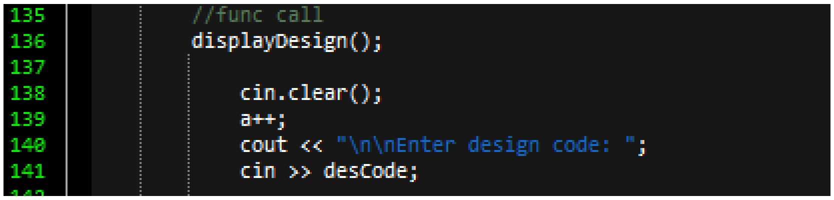
USAGE OF ARRAY

- This program uses array to use a lot of data under one datatype , which is the names of design and the names of model phone.

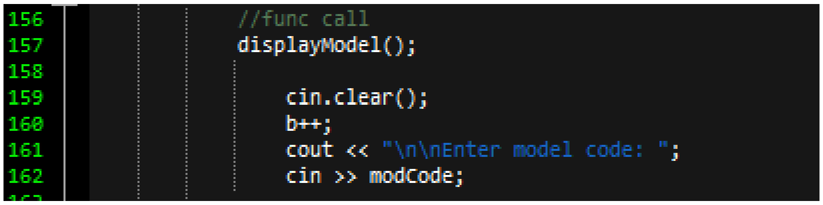


```
//ARRAY DECLARATION

    char tryDesign[100][50], tryModel[100][50];
    int a=0, b=0;
```

- Above are the declaration of the array.

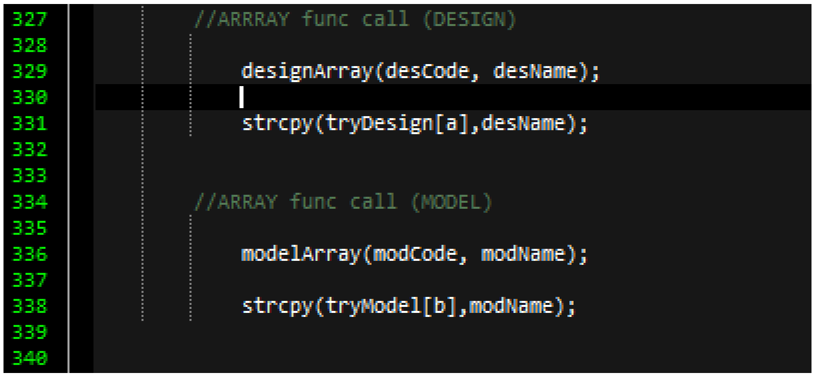


```
        //func call
        displayDesign();

            cin.clear();
            a++;
            cout << "\n\nEnter design code: ";
            cin >> desCode;
```



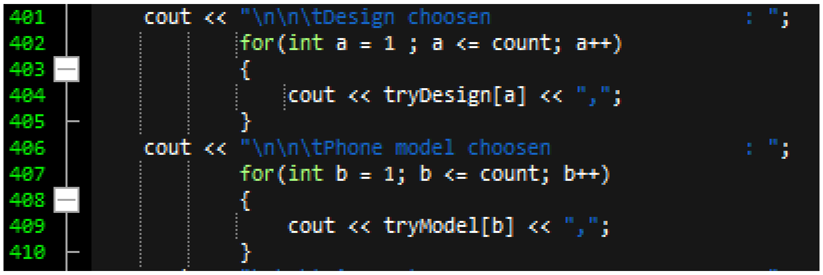
```
            //func call
            displayModel();

                cin.clear();
                b++;
                cout << "\n\nEnter model code: ";
                cin >> modCode;
```

- Above are the iteration of index for array looping. Everytime the design and model is entered it will trigger the iteration.



```
        //ARRRAY func call (DESIGN)

            designArray(desCode, desName);

            strcpy(tryDesign[a],desName);


        //ARRAY func call (MODEL)

            modelArray(modCode, modName);

            strcpy(tryModel[b],modName);

```

- By looking at the full source code(Page A – P) , two functions containing the names of design and model are used and called as above.
- The array will call the names of design and model according to the index which is a and b.



```
    cout << "\n\n\tDesign choosen                 : ";
            for(int a = 1 ; a <= count; a++)
            {
                cout << tryDesign[a] << ",";
            }
    cout << "\n\n\tPhone model choosen            : ";
            for(int b = 1; b <= count; b++)
            {
                cout << tryModel[b] << ",";
            }
```

- At the receipt, a for... looping structure was used to loop the array.
- The array will continuous repeated along with the a/b <= count. COUNT IS THE VALUE OF THE PHONECASES.
- Example: cout << tryDesign[a] is a way to display the names of the design.

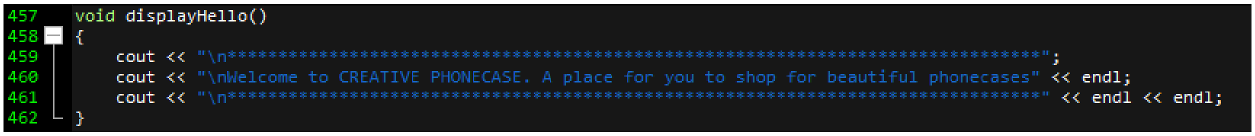
2.1 - LIST OF FUNCTIONS

❖ This program includes the usage of functions to separate some processes.

- **int main()**
- **other functions** for specific purposes

➢ *Functions with specific purposes:*

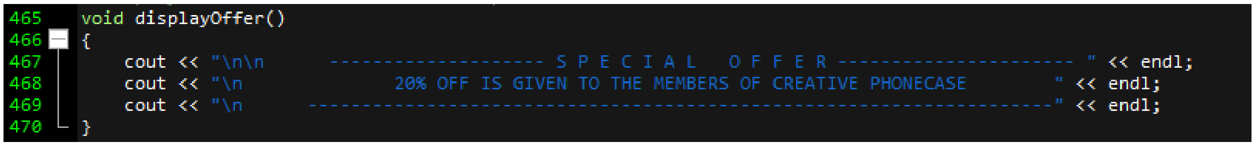
- **Functions without return value:**
- Using **void();**
  - void displayHello()



```
void displayHello()
{
    cout << "\n*************************************************************************************";
    cout << "\nWelcome to CREATIVE PHONECASE. A place for you to shop for beautiful phonecases" << endl;
    cout << "\n*************************************************************************************" << endl << endl;
}
```

This function is used to display a *welcoming text.*

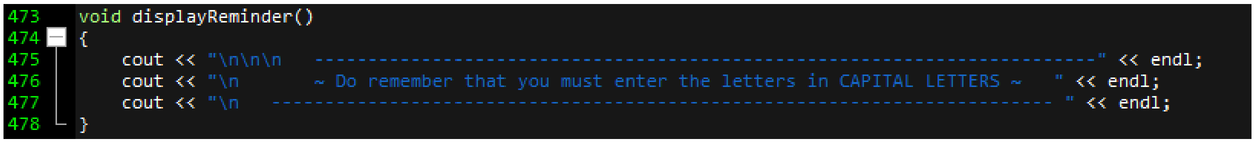
- void displayOffer()



```
void displayOffer()
{
    cout << "\n\n      -------------------- S P E C I A L   O F F E R ---------------------- " << endl;
    cout << "\n            20% OFF IS GIVEN TO THE MEMBERS OF CREATIVE PHONECASE        " << endl;
    cout << "\n     ----------------------------------------------------------------------" << endl;
}
```

This function is used to display text of informing the user that the store will give *20% off* for the member of the store.

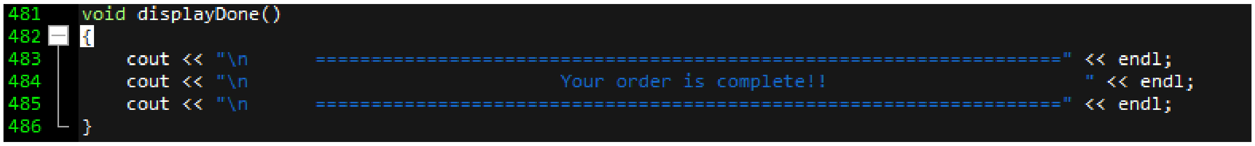
- void displayReminder()



```
void displayReminder()
{
    cout << "\n\n\n   --------------------------------------------------------------------------" << endl;
    cout << "\n       ~ Do remember that you must enter the letters in CAPITAL LETTERS ~   " << endl;
    cout << "\n   -------------------------------------------------------------------------- " << endl;
}
```

This function is used to *display a reminder text* about how to conduct the order.

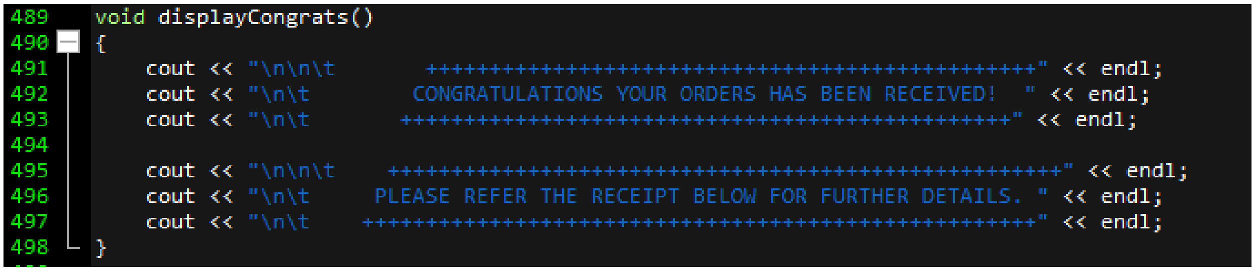
- void displayDone()



```
void displayDone()
{
    cout << "\n      ===================================================================" << endl;
    cout << "\n                              Your order is complete!!                    " << endl;
    cout << "\n      ===================================================================" << endl;
}
```

This function is used to *display the text of a completed order* for one item.

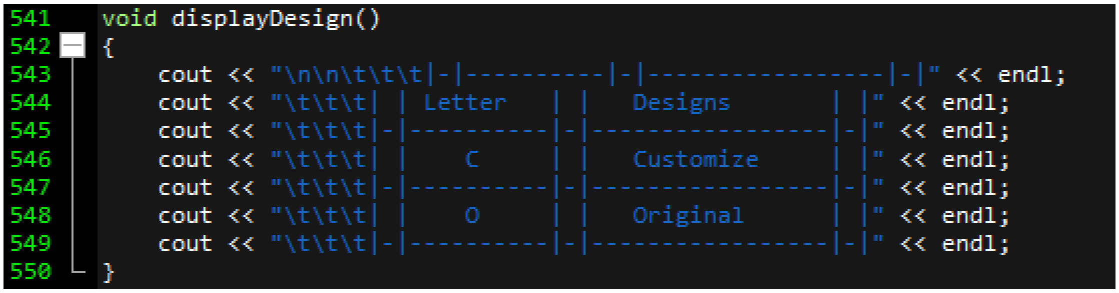
- void displayCongrats()



```
void displayCongrats()
{
    cout << "\n\n\t       ++++++++++++++++++++++++++++++++++++++++++++++++" << endl;
    cout << "\n\t        CONGRATULATIONS YOUR ORDERS HAS BEEN RECEIVED!  " << endl;
    cout << "\n\t       ++++++++++++++++++++++++++++++++++++++++++++++++" << endl;

    cout << "\n\n\t    +++++++++++++++++++++++++++++++++++++++++++++++++++++" << endl;
    cout << "\n\t     PLEASE REFER THE RECEIPT BELOW FOR FURTHER DETAILS. " << endl;
    cout << "\n\t    +++++++++++++++++++++++++++++++++++++++++++++++++++++" << endl;
}
```

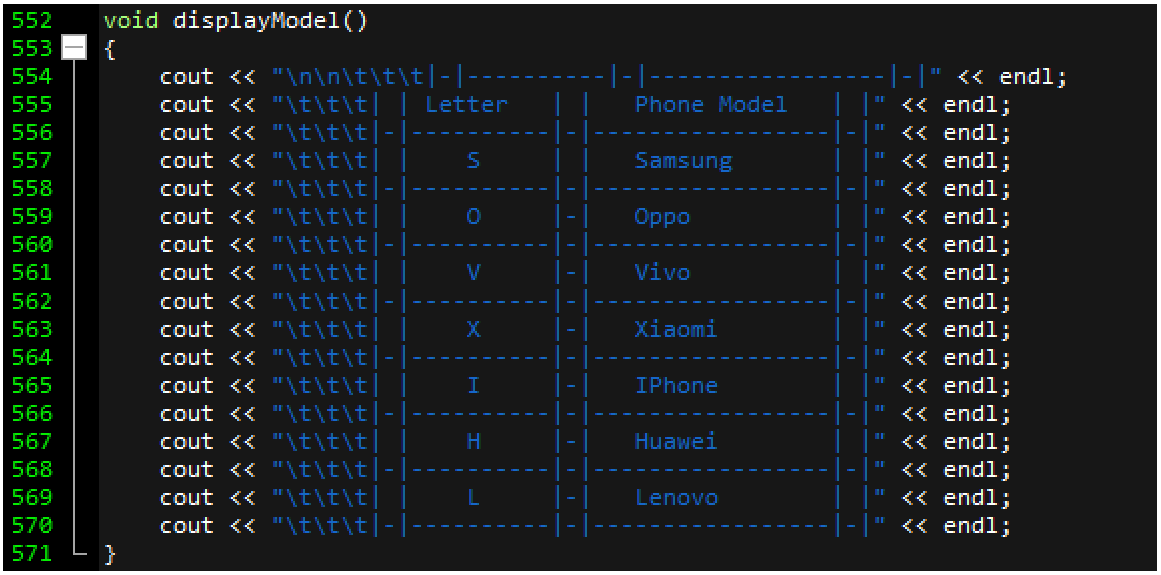
This function is used *to display that the whole order* for one customer is *done*.

- void displayDesign()



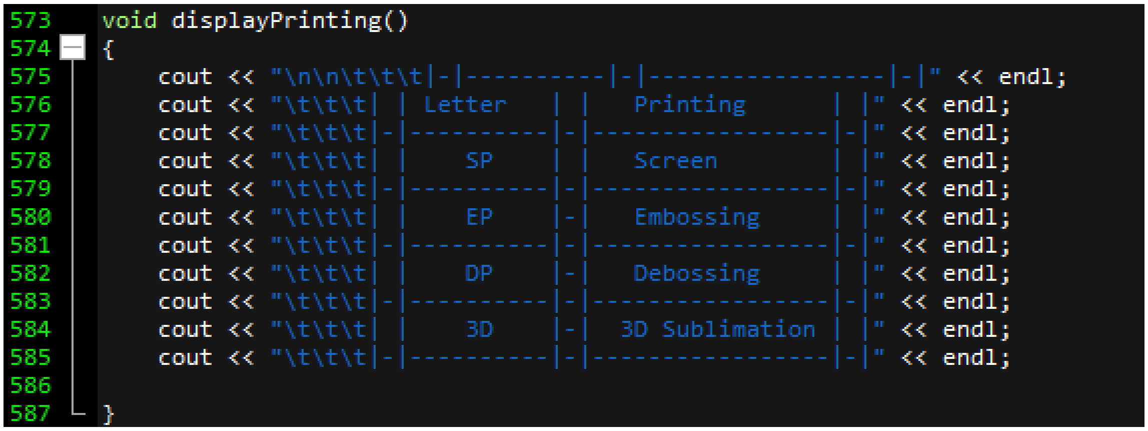
```
void displayDesign()
{
    cout << "\n\n\t\t\t|-|----------|-|-----------------|-|" << endl;
    cout << "\t\t\t| | Letter   | |   Designs       | |" << endl;
    cout << "\t\t\t|-|----------|-|-----------------|-|" << endl;
    cout << "\t\t\t| |    C     | |   Customize     | |" << endl;
    cout << "\t\t\t|-|----------|-|-----------------|-|" << endl;
    cout << "\t\t\t| |    O     | |   Original      | |" << endl;
    cout << "\t\t\t|-|----------|-|-----------------|-|" << endl;
}
```

- void displayModel()



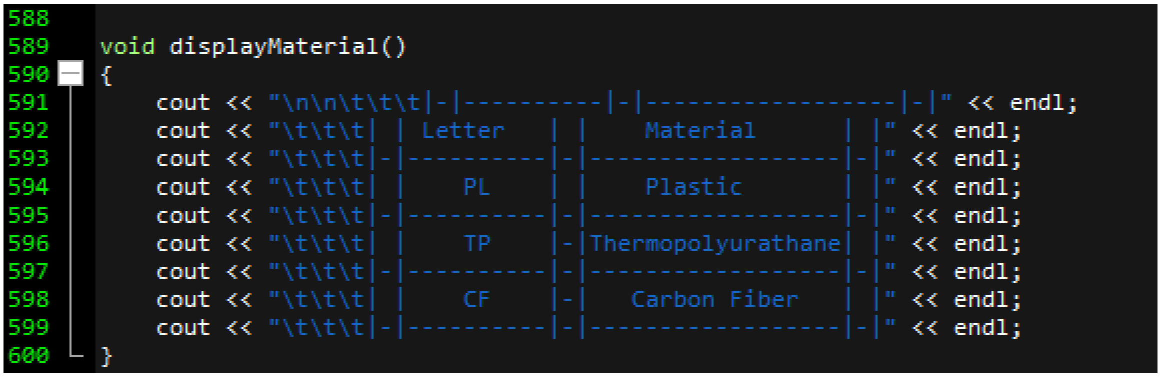
```
void displayModel()
{
    cout << "\n\n\t\t\t|-|----------|-|-----------------|-|" << endl;
    cout << "\t\t\t| | Letter   | |   Phone Model   | |" << endl;
    cout << "\t\t\t|-|----------|-|-----------------|-|" << endl;
    cout << "\t\t\t| |    S     | |   Samsung       | |" << endl;
    cout << "\t\t\t|-|----------|-|-----------------|-|" << endl;
    cout << "\t\t\t| |    O     |-|   Oppo          | |" << endl;
    cout << "\t\t\t|-|----------|-|-----------------|-|" << endl;
    cout << "\t\t\t| |    V     |-|   Vivo          | |" << endl;
    cout << "\t\t\t|-|----------|-|-----------------|-|" << endl;
    cout << "\t\t\t| |    X     |-|   Xiaomi        | |" << endl;
    cout << "\t\t\t|-|----------|-|-----------------|-|" << endl;
    cout << "\t\t\t| |    I     |-|   IPhone        | |" << endl;
    cout << "\t\t\t|-|----------|-|-----------------|-|" << endl;
    cout << "\t\t\t| |    H     |-|   Huawei        | |" << endl;
    cout << "\t\t\t|-|----------|-|-----------------|-|" << endl;
    cout << "\t\t\t| |    L     |-|   Lenovo        | |" << endl;
    cout << "\t\t\t|-|----------|-|-----------------|-|" << endl;
}
```

- void displayPrinting()



```
void displayPrinting()
{
    cout << "\n\n\t\t\t|-|----------|-|------------------|-|" << endl;
    cout << "\t\t\t| | Letter   | |   Printing       | |" << endl;
    cout << "\t\t\t|-|----------|-|------------------|-|" << endl;
    cout << "\t\t\t| |    SP    | |   Screen         | |" << endl;
    cout << "\t\t\t|-|----------|-|------------------|-|" << endl;
    cout << "\t\t\t| |    EP    |-|   Embossing      | |" << endl;
    cout << "\t\t\t|-|----------|-|------------------|-|" << endl;
    cout << "\t\t\t| |    DP    |-|   Debossing      | |" << endl;
    cout << "\t\t\t|-|----------|-|------------------|-|" << endl;
    cout << "\t\t\t| |    3D    |-|   3D Sublimation | |" << endl;
    cout << "\t\t\t|-|----------|-|------------------|-|" << endl;

}
```

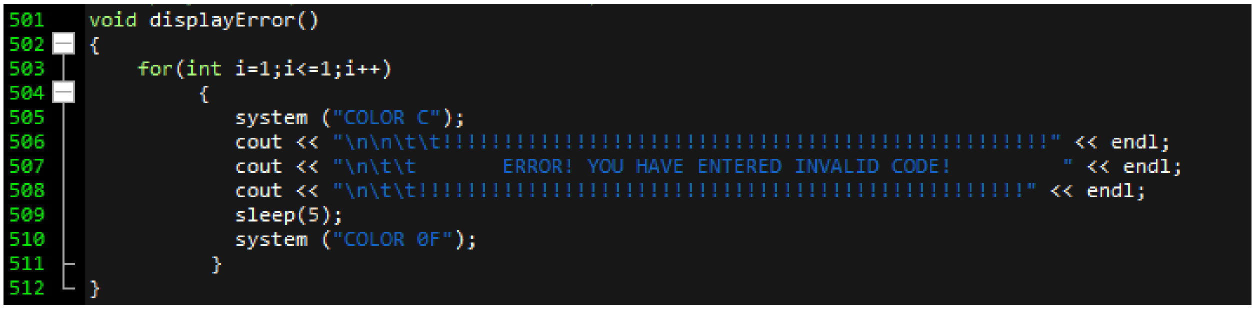
- void displayMaterial()



```

void displayMaterial()
{
    cout << "\n\n\t\t\t|-|----------|-|------------------|-|" << endl;
    cout << "\t\t\t| | Letter   | |    Material      | |" << endl;
    cout << "\t\t\t|-|----------|-|------------------|-|" << endl;
    cout << "\t\t\t| |    PL    | |    Plastic       | |" << endl;
    cout << "\t\t\t|-|----------|-|------------------|-|" << endl;
    cout << "\t\t\t| |    TP    |-|Thermopolyurathane| |" << endl;
    cout << "\t\t\t|-|----------|-|------------------|-|" << endl;
    cout << "\t\t\t| |    CF    |-|   Carbon Fiber   | |" << endl;
    cout << "\t\t\t|-|----------|-|------------------|-|" << endl;
}
```

# These functions are used to *display the menu tables.*

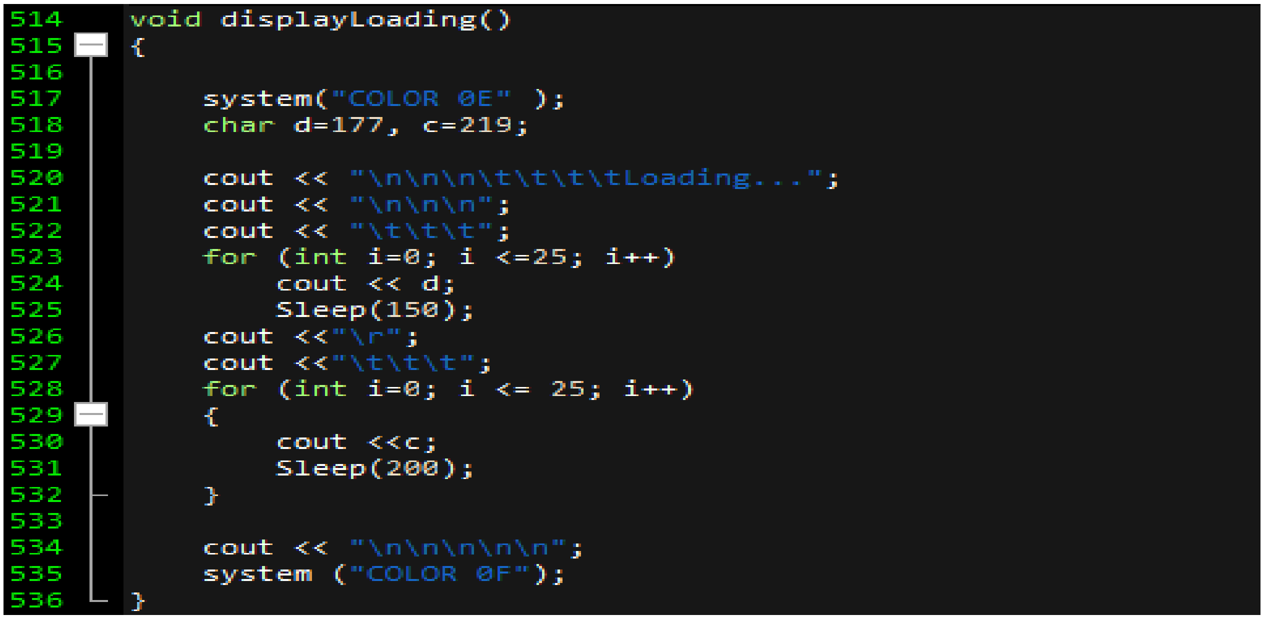
- void displayError()



```
void displayError()
{
    for(int i=1;i<=1;i++)
         {
            system ("COLOR C");
            cout << "\n\n\t\t!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!" << endl;
            cout << "\n\t\t       ERROR! YOU HAVE ENTERED INVALID CODE!         " << endl;
            cout << "\n\t\t!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!" << endl;
            sleep(5);
            system ("COLOR 0F");
          }
}
```

# This function is used to *display error message.*

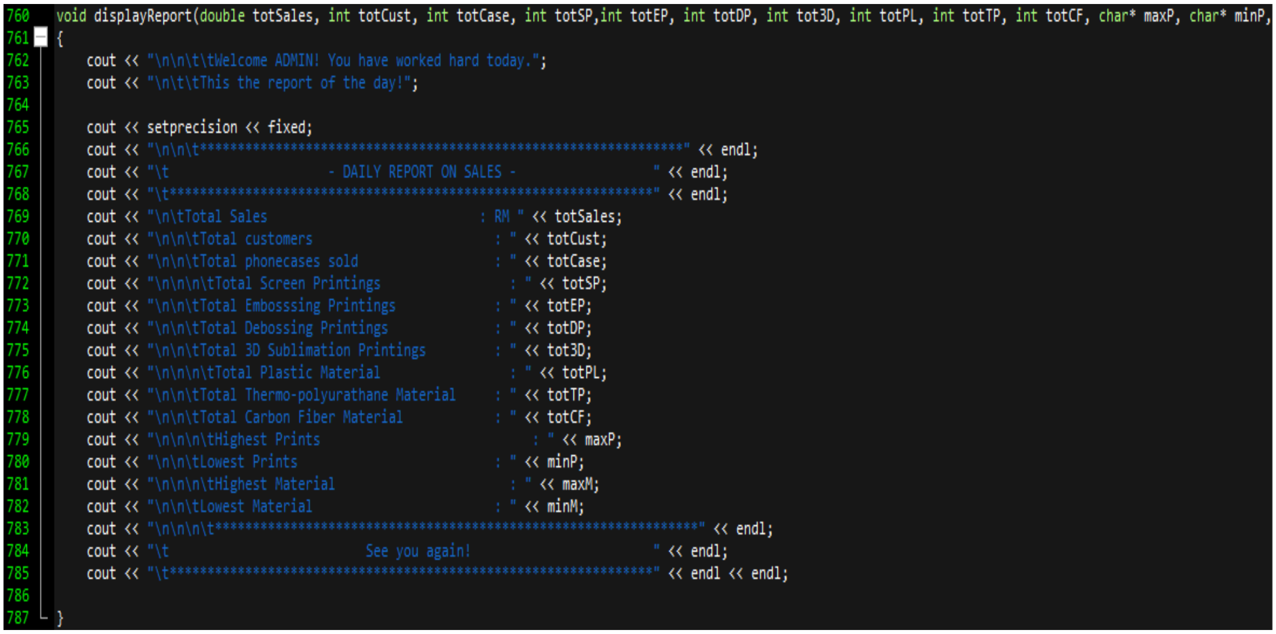
- void displayLoading()



```
void displayLoading()
{

    system("COLOR 0E" );
    char d=177, c=219;

    cout << "\n\n\n\t\t\t\tLoading...";
    cout << "\n\n\n";
    cout << "\t\t\t";
    for (int i=0; i <=25; i++)
        cout << d;
        Sleep(150);
    cout <<"\r";
    cout <<"\t\t\t";
    for (int i=0; i <= 25; i++)
    {
        cout <<c;
        Sleep(200);
    }

    cout << "\n\n\n\n\n";
    system ("COLOR 0F");
}
```

This function is used to *display a loading art work.*

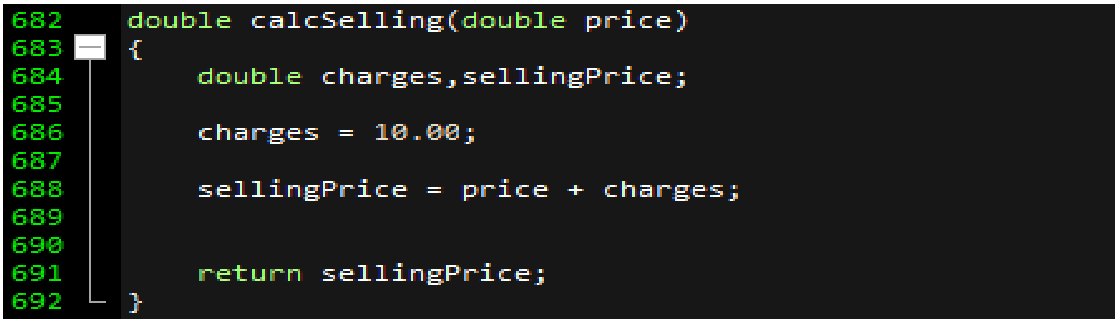
- void displayReport(double, int, int , int ,int, int , int, int, int, int, char*, char*, char*, char*)



```
void displayReport(double totSales, int totCust, int totCase, int totSP,int totEP, int totDP, int tot3D, int totPL, int totTP, int totCF, char* maxP, char* minP,
{
    cout << "\n\n\t\tWelcome ADMIN! You have worked hard today.";
    cout << "\n\t\tThis the report of the day!";

    cout << setprecision << fixed;
    cout << "\n\n\t*****************************************************************" << endl;
    cout << "\t                     - DAILY REPORT ON SALES -                  " << endl;
    cout << "\t*****************************************************************" << endl;
    cout << "\n\tTotal Sales                             : RM " << totSales;
    cout << "\n\n\tTotal customers                         : " << totCust;
    cout << "\n\n\tTotal phonecases sold                   : " << totCase;
    cout << "\n\n\n\tTotal Screen Printings                  : " << totSP;
    cout << "\n\n\tTotal Embosssing Printings              : " << totEP;
    cout << "\n\n\tTotal Debossing Printings               : " << totDP;
    cout << "\n\n\tTotal 3D Sublimation Printings          : " << tot3D;
    cout << "\n\n\n\tTotal Plastic Material                  : " << totPL;
    cout << "\n\n\tTotal Thermo-polyurathane Material      : " << totTP;
    cout << "\n\n\tTotal Carbon Fiber Material             : " << totCF;
    cout << "\n\n\n\tHighest Prints                             : " << maxP;
    cout << "\n\n\tLowest Prints                           : " << minP;
    cout << "\n\n\n\tHighest Material                        : " << maxM;
    cout << "\n\n\tLowest Material                         : " << minM;
    cout << "\n\n\n\t*****************************************************************" << endl;
    cout << "\t                          See you again!                        " << endl;
    cout << "\t*****************************************************************" << endl << endl;

}
```

This function is used *to display the daily report.*

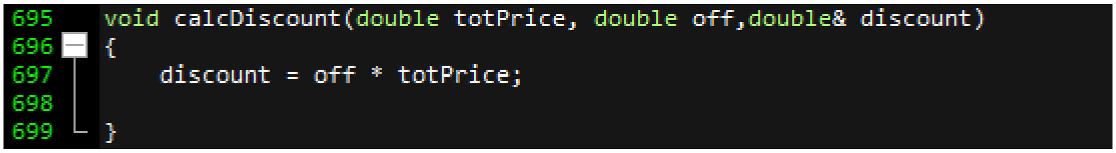
- **Functions with return value:**
  - **Double()**
    - double calcSelling(double price);



```
double calcSelling(double price)
{
    double charges,sellingPrice;

    charges = 10.00;

    sellingPrice = price + charges;


    return sellingPrice;
}
```

This function is used *to calculate the selling price* of one item and return the value of the calculation.

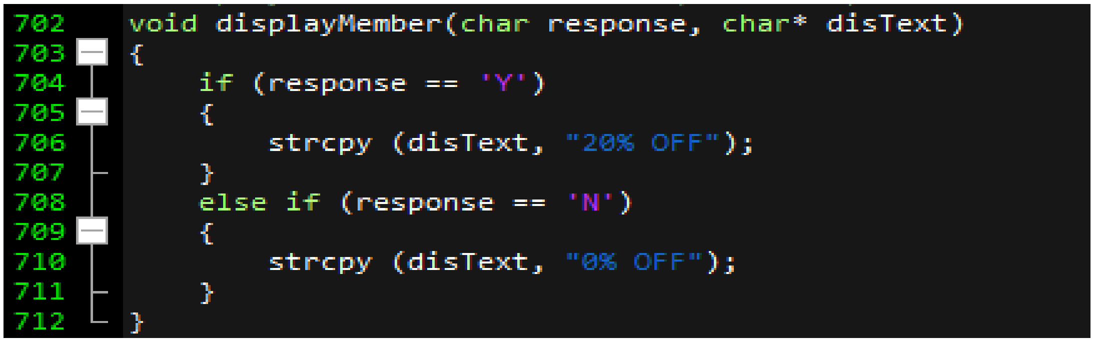
- **Functions with reference parameter**
  - **Void()**
    - void calcDiscount(double totPrice, double off,double& discount);



```
void calcDiscount(double totPrice, double off,double& discount)
{
    discount = off * totPrice;

}
```

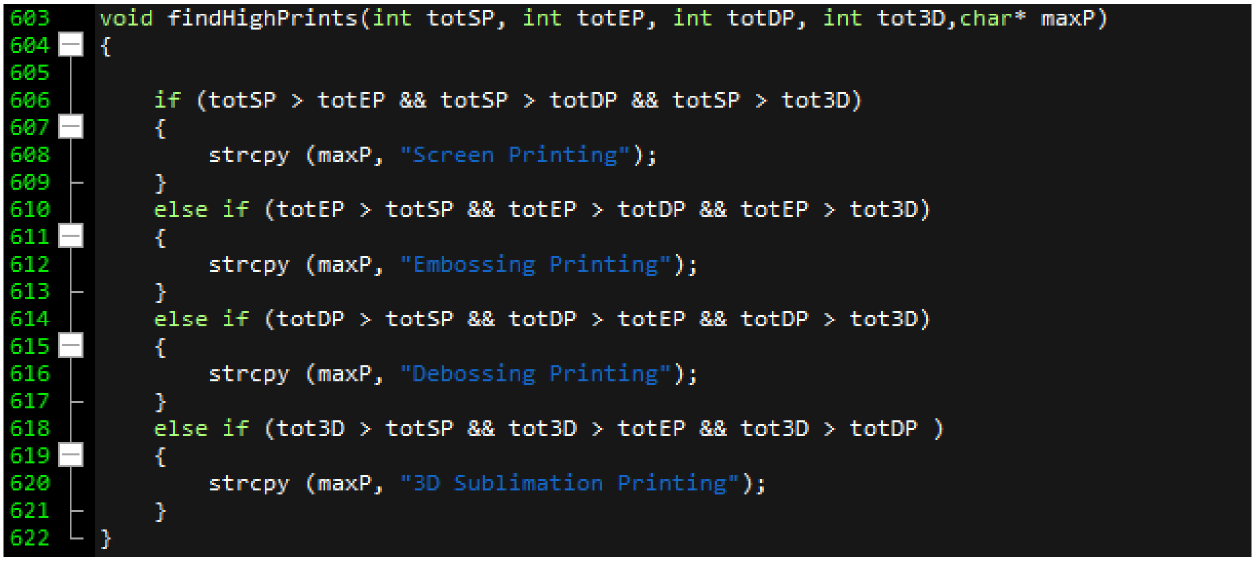
This function is used to *calculate the discount* from the total price of one customer as it minus the discount from one purchase. The discount price is *brought up as a reference*.

    - void displayMember(char response, char* disText)



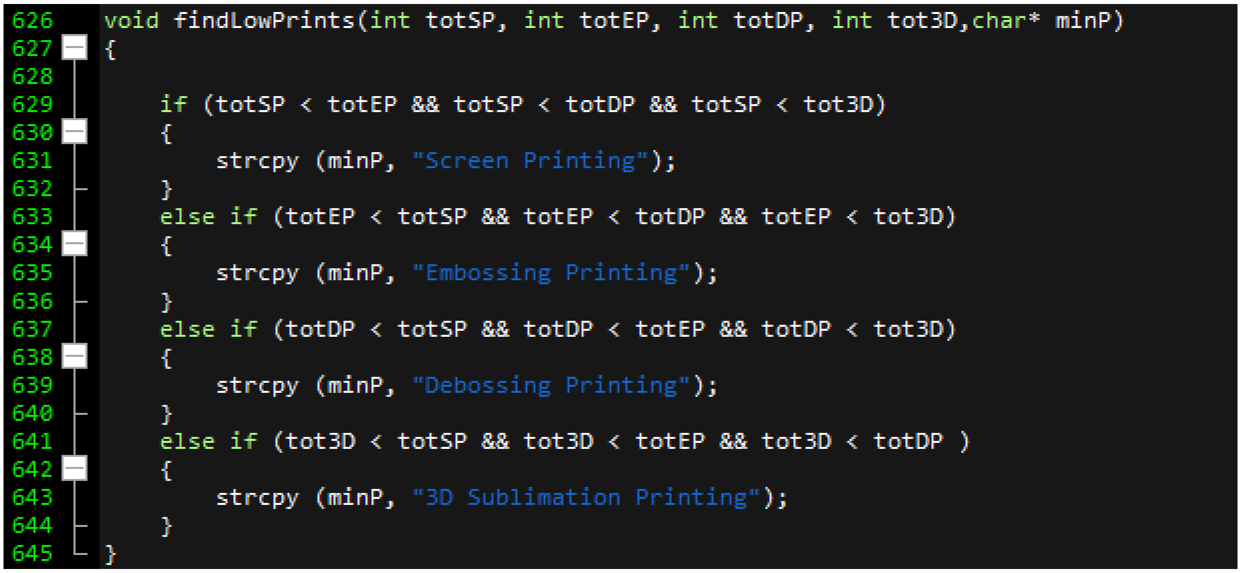
```
void displayMember(char response, char* disText)
{
    if (response == 'Y')
    {
        strcpy (disText, "20% OFF");
    }
    else if (response == 'N')
    {
        strcpy (disText, "0% OFF");
    }
}
```

- void findHighPrints(int totSP, int totEP, int totDP, int tot3D,char* maxP)



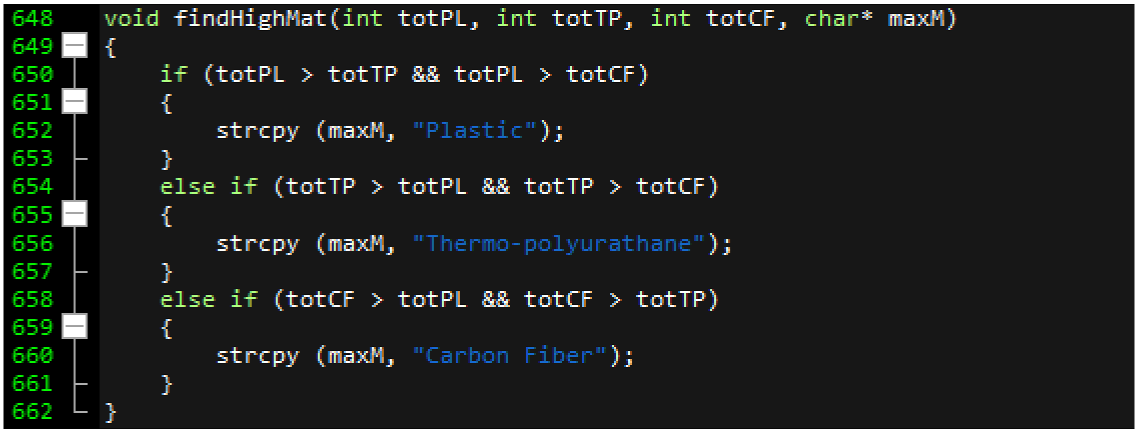
```
void findHighPrints(int totSP, int totEP, int totDP, int tot3D,char* maxP)
{

    if (totSP > totEP && totSP > totDP && totSP > tot3D)
    {
        strcpy (maxP, "Screen Printing");
    }
    else if (totEP > totSP && totEP > totDP && totEP > tot3D)
    {
        strcpy (maxP, "Embossing Printing");
    }
    else if (totDP > totSP && totDP > totEP && totDP > tot3D)
    {
        strcpy (maxP, "Debossing Printing");
    }
    else if (tot3D > totSP && tot3D > totEP && tot3D > totDP )
    {
        strcpy (maxP, "3D Sublimation Printing");
    }
}
```

- void findLowPrints(int totSP, int totEP, int totDP, int tot3D,char* minP)



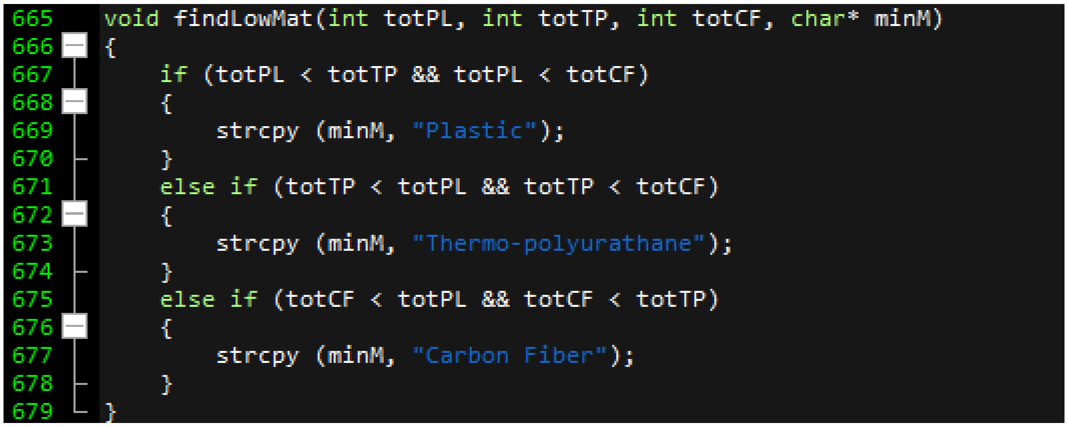
```
void findLowPrints(int totSP, int totEP, int totDP, int tot3D,char* minP)
{

    if (totSP < totEP && totSP < totDP && totSP < tot3D)
    {
        strcpy (minP, "Screen Printing");
    }
    else if (totEP < totSP && totEP < totDP && totEP < tot3D)
    {
        strcpy (minP, "Embossing Printing");
    }
    else if (totDP < totSP && totDP < totEP && totDP < tot3D)
    {
        strcpy (minP, "Debossing Printing");
    }
    else if (tot3D < totSP && tot3D < totEP && tot3D < totDP )
    {
        strcpy (minP, "3D Sublimation Printing");
    }
}
```

- void findHighPrints(int totSP, int totEP, int totDP, int tot3D,char* maxP)



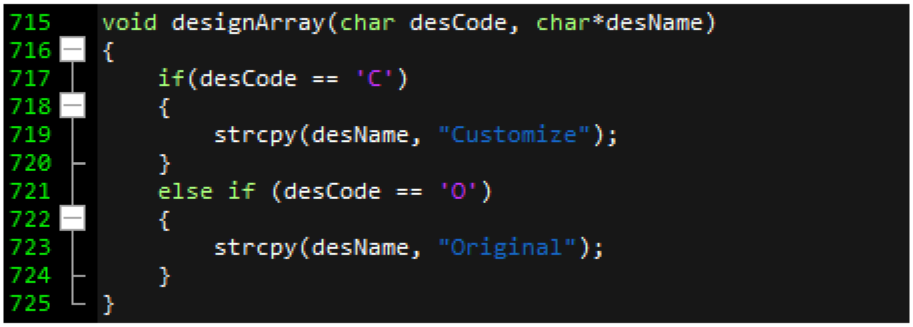
```
void findHighMat(int totPL, int totTP, int totCF, char* maxM)
{
    if (totPL > totTP && totPL > totCF)
    {
        strcpy (maxM, "Plastic");
    }
    else if (totTP > totPL && totTP > totCF)
    {
        strcpy (maxM, "Thermo-polyurathane");
    }
    else if (totCF > totPL && totCF > totTP)
    {
        strcpy (maxM, "Carbon Fiber");
    }
}
```

- void findLowMat(int totPL, int totTP, int totCF, char* minM)



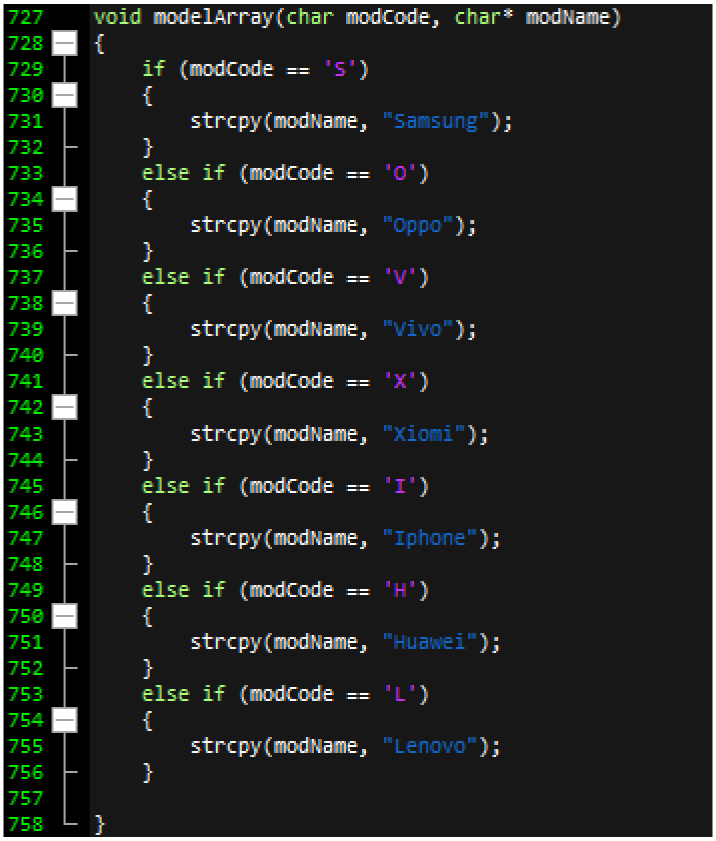
```
void findLowMat(int totPL, int totTP, int totCF, char* minM)
{
    if (totPL < totTP && totPL < totCF)
    {
        strcpy (minM, "Plastic");
    }
    else if (totTP < totPL && totTP < totCF)
    {
        strcpy (minM, "Thermo-polyurathane");
    }
    else if (totCF < totPL && totCF < totTP)
    {
        strcpy (minM, "Carbon Fiber");
    }
}
```

- void designArray(char desCode, char*desName)



```
void designArray(char desCode, char*desName)
{
    if(desCode == 'C')
    {
        strcpy(desName, "Customize");
    }
    else if (desCode == 'O')
    {
        strcpy(desName, "Original");
    }
}
```

- void modelArray(char modCode, char* modName)

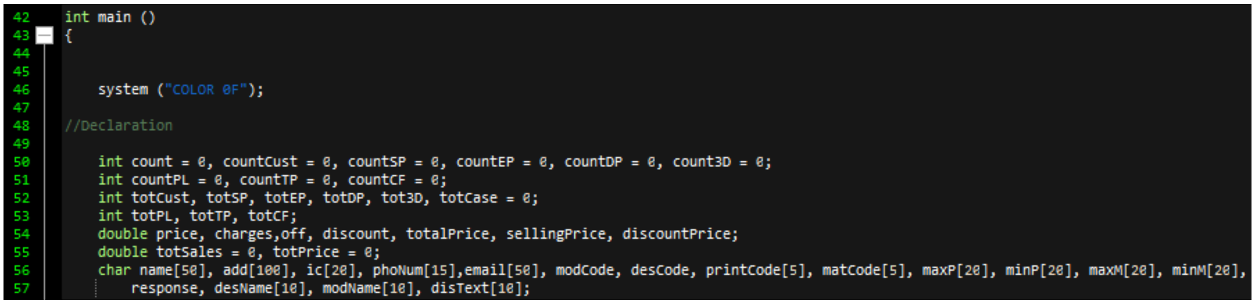


```
void modelArray(char modCode, char* modName)
{
    if (modCode == 'S')
    {
        strcpy(modName, "Samsung");
    }
    else if (modCode == 'O')
    {
        strcpy(modName, "Oppo");
    }
    else if (modCode == 'V')
    {
        strcpy(modName, "Vivo");
    }
    else if (modCode == 'X')
    {
        strcpy(modName, "Xiomi");
    }
    else if (modCode == 'I')
    {
        strcpy(modName, "Iphone");
    }
    else if (modCode == 'H')
    {
        strcpy(modName, "Huawei");
    }
    else if (modCode == 'L')
    {
        strcpy(modName, "Lenovo");
    }

}
```

This function is used to the *copy texts* and used *to display it in receipt*.

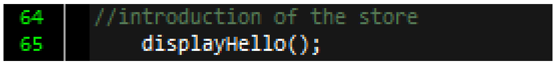
int main() function

- **int main()** is used as the main function where all of the other functions is being called and ordered to do their purposes in the body of the int main().

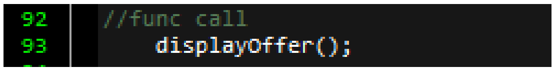


```
int main ()
{


    system ("COLOR 0F");

//Declaration

    int count = 0, countCust = 0, countSP = 0, countEP = 0, countDP = 0, count3D = 0;
    int countPL = 0, countTP = 0, countCF = 0;
    int totCust, totSP, totEP, totDP, tot3D, totCase = 0;
    int totPL, totTP, totCF;
    double price, charges,off, discount, totalPrice, sellingPrice, discountPrice;
    double totSales = 0, totPrice = 0;
    char name[50], add[100], ic[20], phoNum[15],email[50], modCode, desCode, printCode[5], matCode[5], maxP[20], minP[20], maxM[20], minM[20],
        response, desName[10], modName[10], disText[10];
```

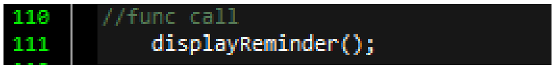
- Function call of the **functions without return value:**



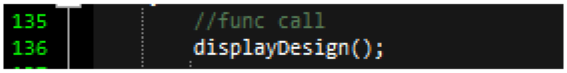
```
//introduction of the store
    displayHello();
```



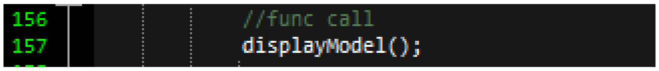
```
//func call
    displayOffer();
```



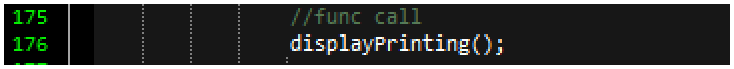
```
//func call
    displayReminder();
```



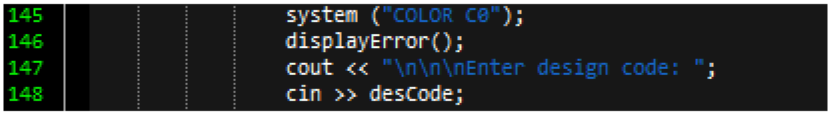
```
//func call
displayDesign();
```



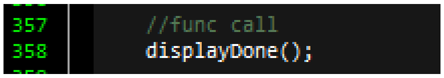
```
//func call
displayModel();
```



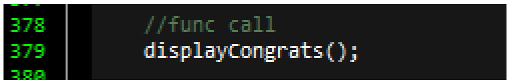
```
//func call
displayPrinting();
```



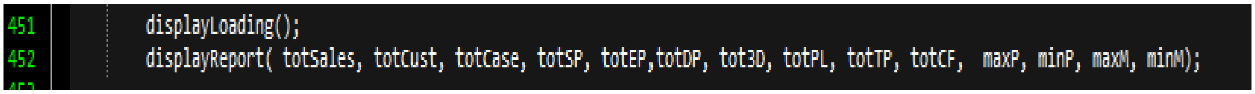
```
system ("COLOR C0");
displayError();
cout << "\n\n\nEnter design code: ";
cin >> desCode;
```



```
//func call
displayDone();
```

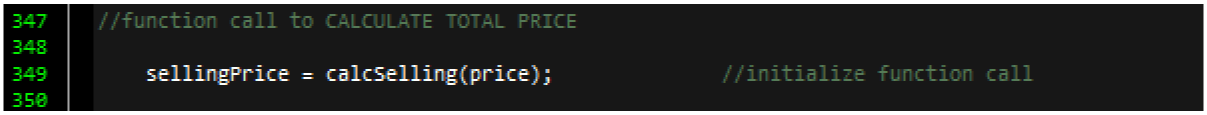


```
//func call
displayCongrats();
```



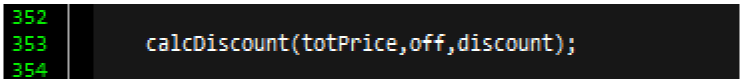
```
displayLoading();
displayReport( totSales, totCust, totCase, totSP, totEP,totDP, tot3D, totPL, totTP, totCF,  maxP, minP, maxM, minM);
```

- Function call of the **functions with return value:**

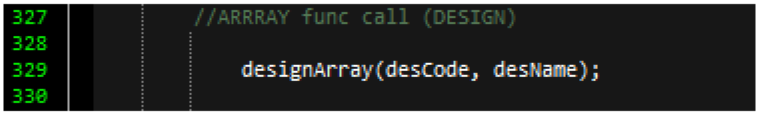


```
//function call to CALCULATE TOTAL PRICE

    sellingPrice = calcSelling(price);              //initialize function call
```

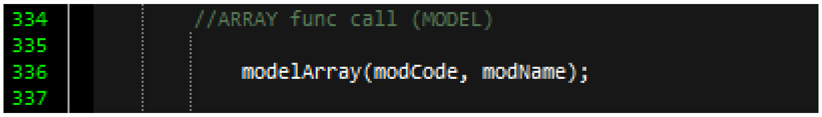
- Function call of the **functions with reference parameter**



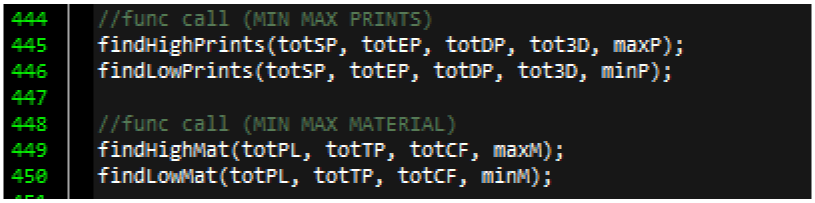
```
    calcDiscount(totPrice,off,discount);
```



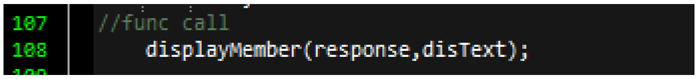
```
        //ARRRAY func call (DESIGN)

            designArray(desCode, desName);
```



```
        //ARRAY func call (MODEL)

            modelArray(modCode, modName);
```



```
//func call (MIN MAX PRINTS)
findHighPrints(totSP, totEP, totDP, tot3D, maxP);
findLowPrints(totSP, totEP, totDP, tot3D, minP);

//func call (MIN MAX MATERIAL)
findHighMat(totPL, totTP, totCF, maxM);
findLowMat(totPL, totTP, totCF, minM);
```

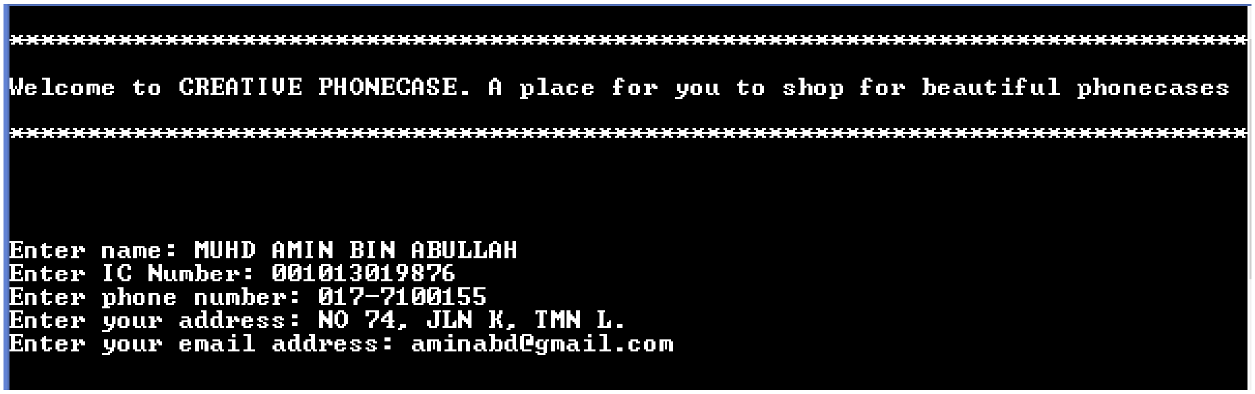


```
//func call
    displayMember(response,disText);
```

3.0 – PROGRAM TESTING

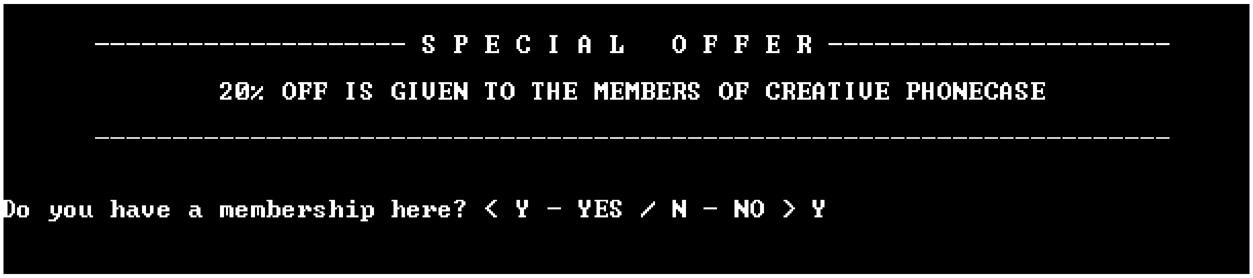
- These are the steps and the flow of the program:

# STEP 1: The user will enter their *personal information* as the first input.



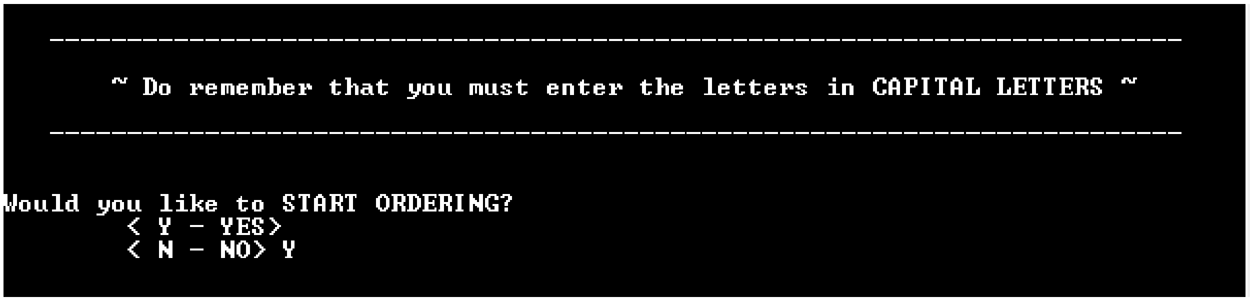
```
********************************************************************************
Welcome to CREATIVE PHONECASE. A place for you to shop for beautiful phonecases
********************************************************************************



Enter name: MUHD AMIN BIN ABULLAH
Enter IC Number: 001013019876
Enter phone number: 017-7100155
Enter your address: NO 74, JLN K, TMN L.
Enter your email address: aminabd@gmail.com
```

# STEP 2: A text of offer will displayed . The user will enter *if they are a member*. In this case, Yes.



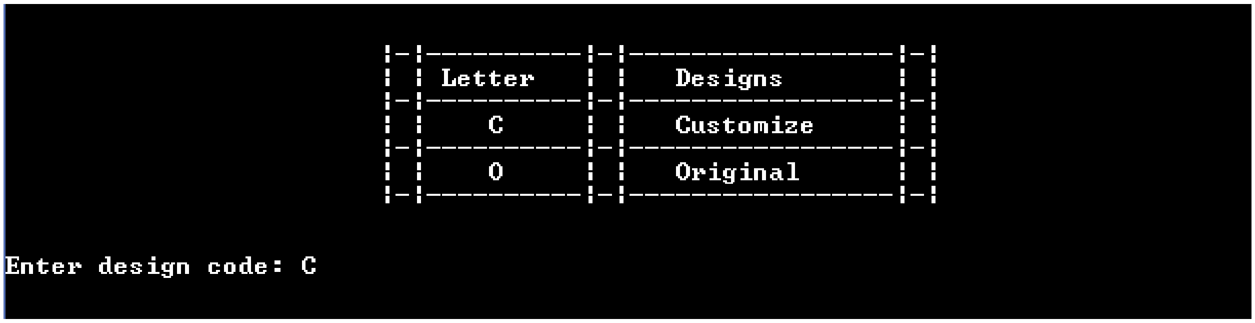
```
     -------------------- S P E C I A L   O F F E R ----------------------

             20% OFF IS GIVEN TO THE MEMBERS OF CREATIVE PHONECASE

     ---------------------------------------------------------------------


Do you have a membership here? < Y - YES / N - NO > Y
```

# STEP 3: A text of a reminder will displayed. Then, user will enter either they want to *start ordering* or not. In this case, Yes.



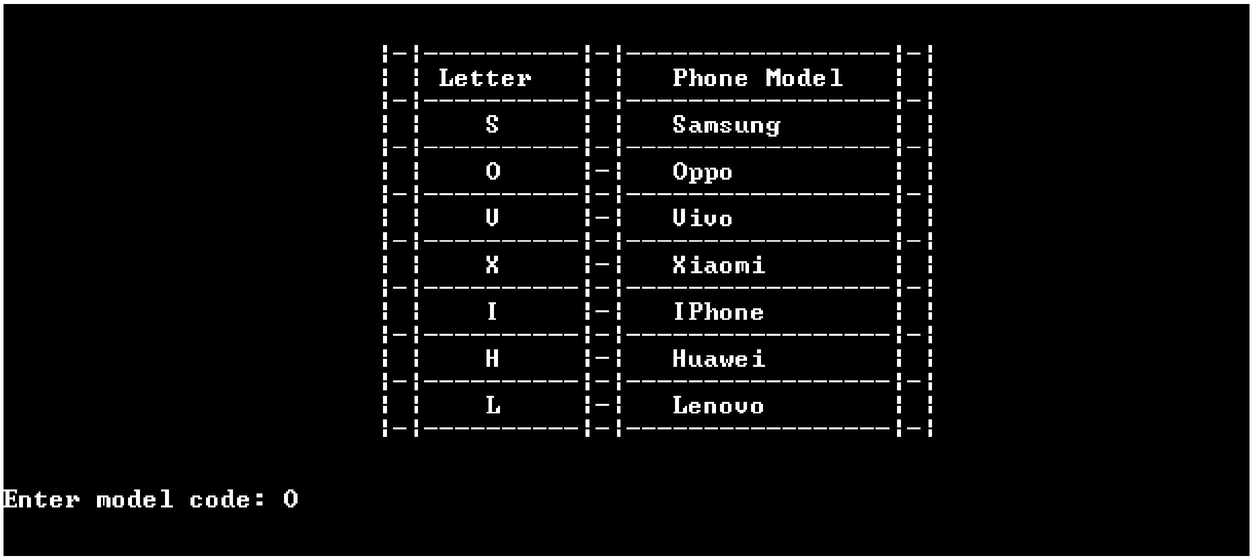
```
   -------------------------------------------------------------------------

       ~ Do remember that you must enter the letters in CAPITAL LETTERS ~

   -------------------------------------------------------------------------


Would you like to START ORDERING?
        < Y - YES>
        < N - NO> Y
```

# STEP 4: Menu table will be displayed. The user will *enter the design code.*



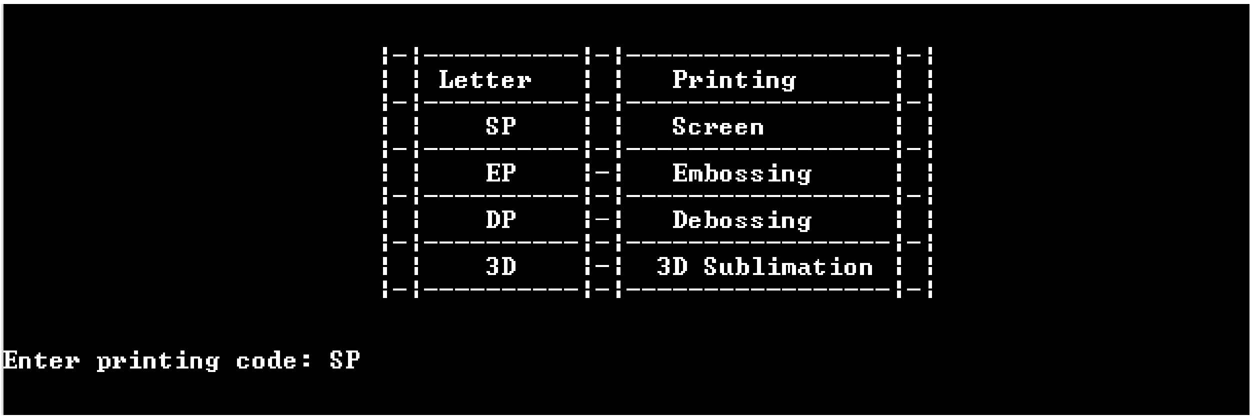
```
|-|-----------|-|------------------|-|
| | Letter    | |   Designs        | |
|-|-----------|-|------------------|-|
| |     C     | |   Customize      | |
|-|-----------|-|------------------|-|
| |     O     | |   Original       | |
|-|-----------|-|------------------|-|


Enter design code: C
```

# STEP 5: Menu table will be displayed. The user will *enter the model code*.



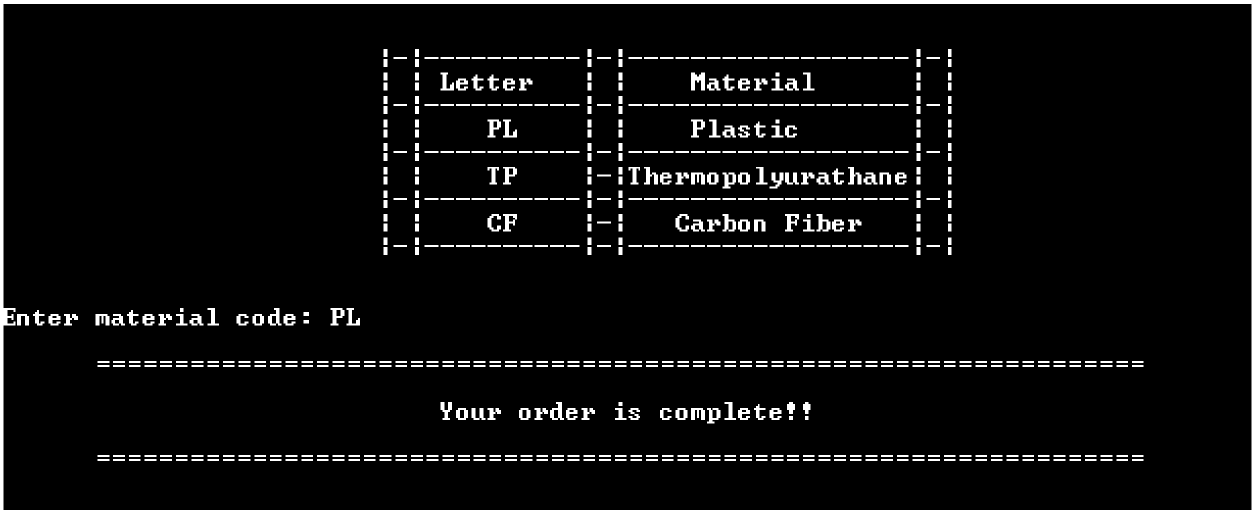
```
|-|-----------|-|------------------|-|
| | Letter    | |   Phone Model    | |
|-|-----------|-|------------------|-|
| |     S     | |   Samsung        | |
|-|-----------|-|------------------|-|
| |     O     |-|   Oppo           | |
|-|-----------|-|------------------|-|
| |     V     |-|   Vivo           | |
|-|-----------|-|------------------|-|
| |     X     |-|   Xiaomi         | |
|-|-----------|-|------------------|-|
| |     I     |-|   IPhone         | |
|-|-----------|-|------------------|-|
| |     H     |-|   Huawei         | |
|-|-----------|-|------------------|-|
| |     L     |-|   Lenovo         | |
|-|-----------|-|------------------|-|


Enter model code: O
```

# STEP 6: Menu table will be displayed. The user will *enter the printing code*.



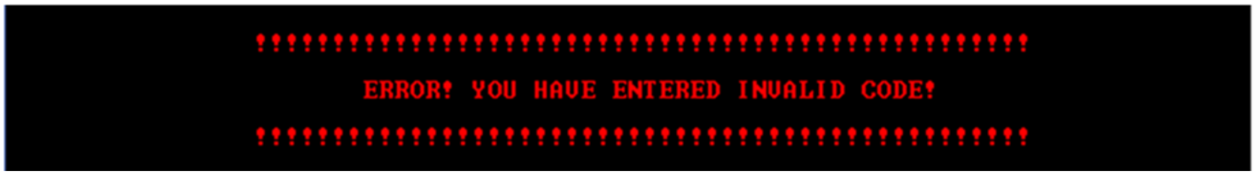
```
|-|-----------|-|------------------|-|
| | Letter    | |   Printing       | |
|-|-----------|-|------------------|-|
| |     SP    | |   Screen         | |
|-|-----------|-|------------------|-|
| |     EP    |-|   Embossing      | |
|-|-----------|-|------------------|-|
| |     DP    |-|   Debossing      | |
|-|-----------|-|------------------|-|
| |     3D    |-|  3D Sublimation  | |
|-|-----------|-|------------------|-|


Enter printing code: SP
```

# STEP 7: Menu table will be displayed. The user will *enter the material code.* A message of order completed will be displyed as well.



```
                        ¦-¦-----------¦-¦-------------------¦-¦
                        ¦ ¦  Letter   ¦ ¦     Material      ¦ ¦
                        ¦-¦-----------¦-¦-------------------¦-¦
                        ¦ ¦    PL     ¦ ¦     Plastic       ¦ ¦
                        ¦-¦-----------¦-¦-------------------¦-¦
                        ¦ ¦    TP     ¦-¦Thermopolyurathane ¦ ¦
                        ¦-¦-----------¦-¦-------------------¦-¦
                        ¦ ¦    CF     ¦-¦   Carbon Fiber    ¦ ¦
                        ¦-¦-----------¦-¦-------------------¦-¦


Enter material code: PL
      ====================================================================

                      Your order is complete!!

      ====================================================================
```

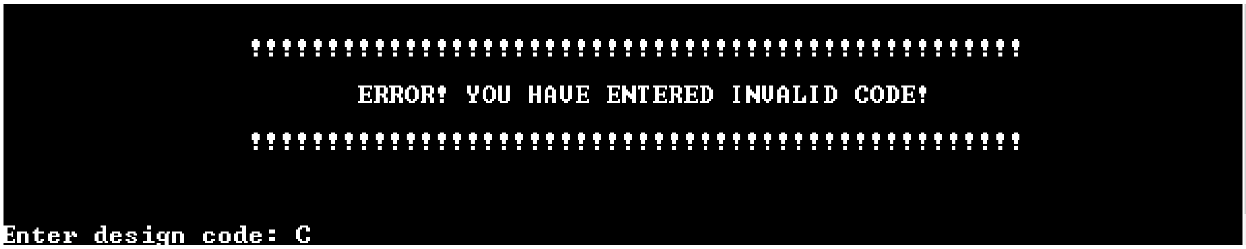
# STEP EXTRA : But if the user *entered a wrong input* in any of the questions asked. An *error message* will be displayed.



```
                !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

                       ERROR! YOU HAVE ENTERED INVALID CODE!

                !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
```

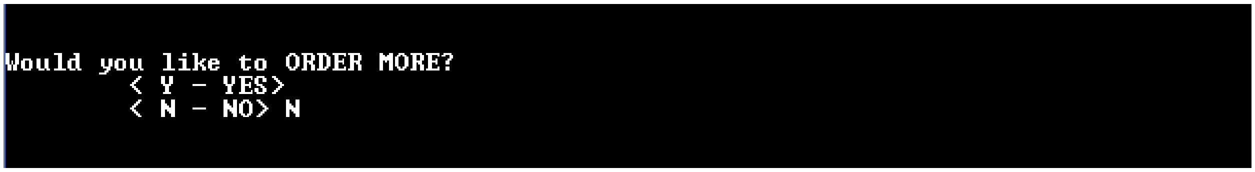
The error message will blink the color from red font to white.

The user will asked to *re-enter the design code* according to the table.



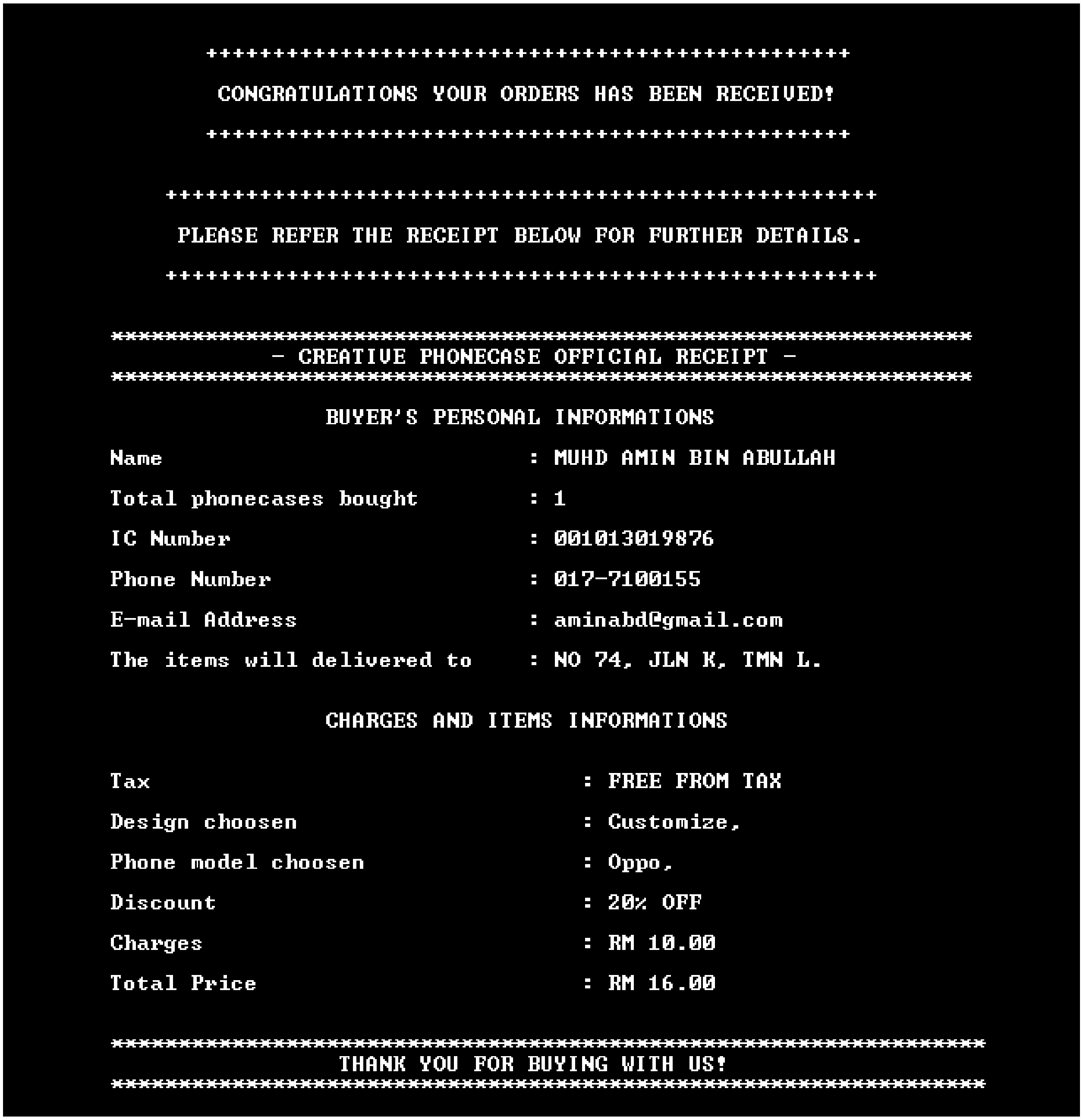
```
                !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

                       ERROR! YOU HAVE ENTERED INVALID CODE!

                !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!


Enter design code: C
```

# STEP 8: A question will be asked as below shown. The user will decide if they want to *order more or no.* In this case the user *entered N (No).*



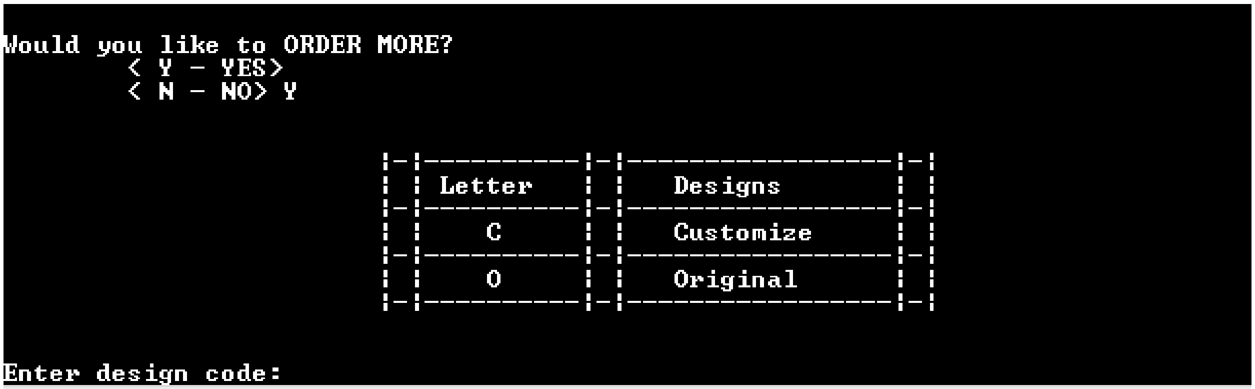
```
Would you like to ORDER MORE?
        < Y - YES>
        < N - NO> N
```

This will resulted the display of congratulation message and *displayed the receipt* as below.



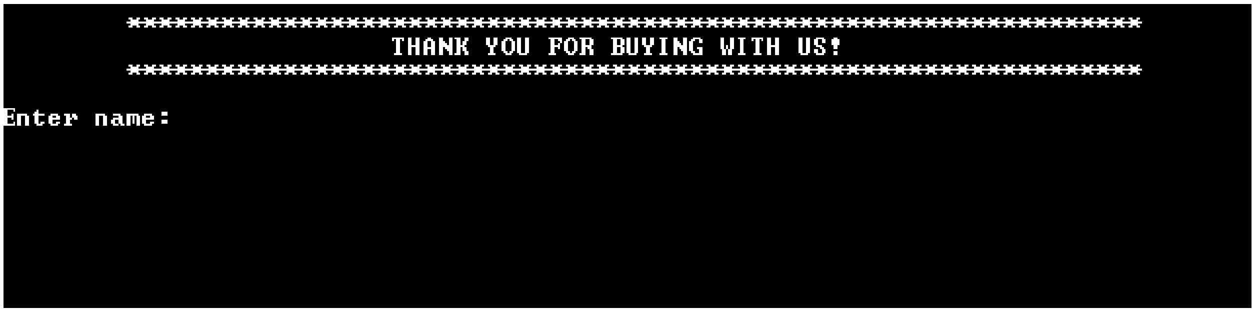
```
               ++++++++++++++++++++++++++++++++++++++++++++++++

                CONGRATULATIONS YOUR ORDERS HAS BEEN RECEIVED!

               ++++++++++++++++++++++++++++++++++++++++++++++++


            +++++++++++++++++++++++++++++++++++++++++++++++++++++

             PLEASE REFER THE RECEIPT BELOW FOR FURTHER DETAILS.

            +++++++++++++++++++++++++++++++++++++++++++++++++++++


        *****************************************************************
                    - CREATIVE PHONECASE OFFICIAL RECEIPT -
        *****************************************************************

                        BUYER'S PERSONAL INFORMATIONS

        Name                           : MUHD AMIN BIN ABULLAH

        Total phonecases bought        : 1

        IC Number                      : 001013019876

        Phone Number                   : 017-7100155

        E-mail Address                 : aminabd@gmail.com

        The items will delivered to    : NO 74, JLN K, TMN L.

                        CHARGES AND ITEMS INFORMATIONS


        Tax                                : FREE FROM TAX

        Design choosen                     : Customize,

        Phone model choosen                : Oppo,

        Discount                           : 20% OFF

        Charges                            : RM 10.00

        Total Price                        : RM 16.00


        ******************************************************************
                         THANK YOU FOR BUYING WITH US!
        ******************************************************************
```

# STEP EXTRA : But if the user *entered Y (Yes).* The program will loop, started from **#STEP 4 UNTIL #STEP 8.**



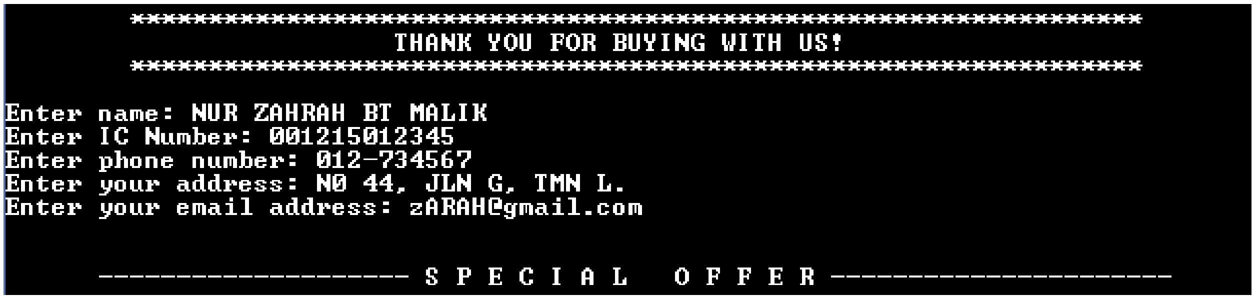
```
Would you like to ORDER MORE?
        < Y - YES>
        < N - NO> Y


                        |-|-----------|-|-----------------|-|
                        | | Letter    | |   Designs       | |
                        |-|-----------|-|-----------------|-|
                        | |     C     | |   Customize     | |
                        |-|-----------|-|-----------------|-|
                        | |     O     | |   Original      | |
                        |-|-----------|-|-----------------|-|


Enter design code:
```

# STEP 9: After the user get the receipt, it will *asked for name*. The next customer will enter *their name* the *process will loop* from **#STEP 1 UNTILL #STEP8.**



```
        *****************************************************************
                         THANK YOU FOR BUYING WITH US!
        *****************************************************************

Enter name:
```

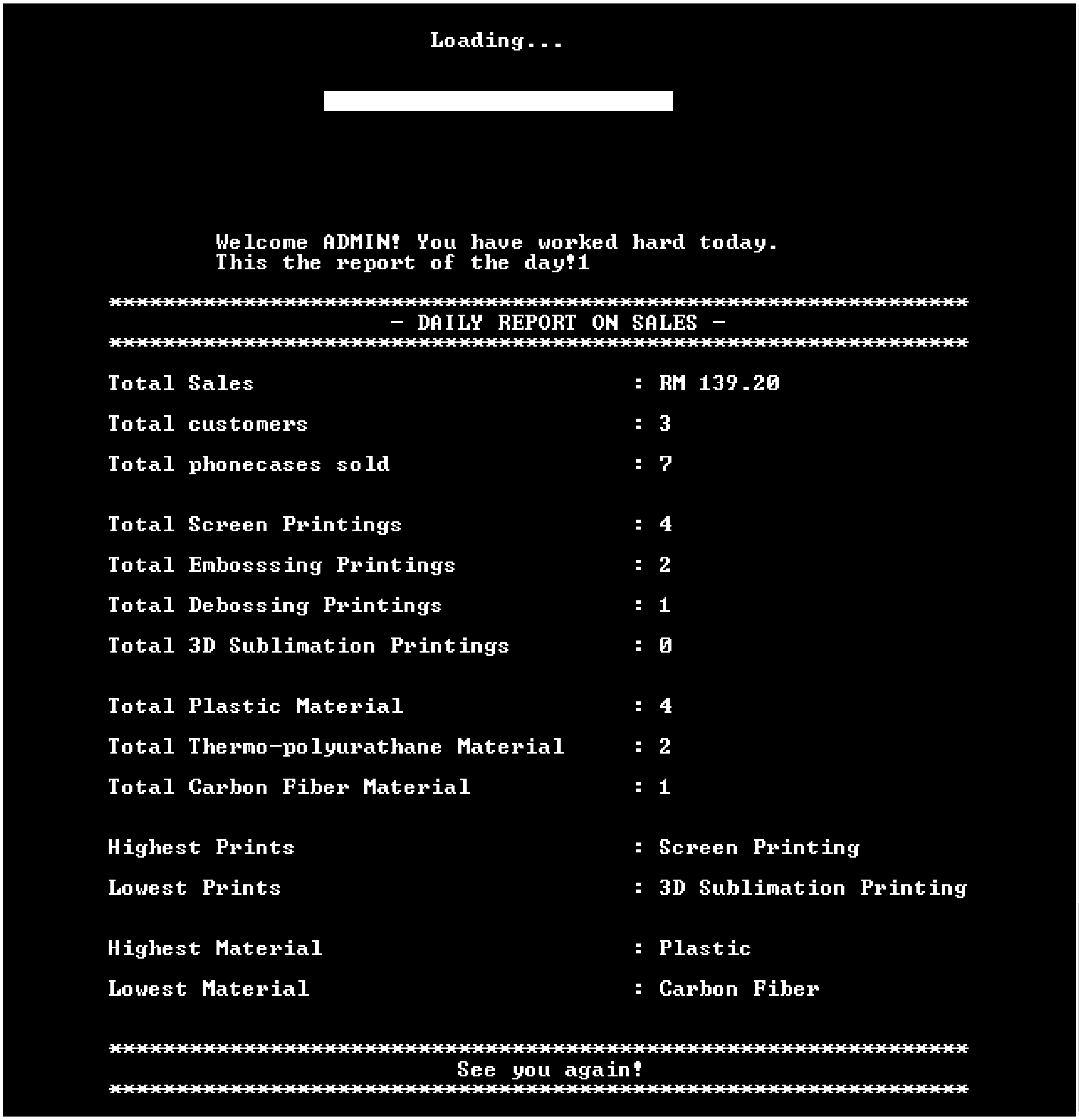
Example next customer's information fill:



```
        *****************************************************************
                         THANK YOU FOR BUYING WITH US!
        *****************************************************************

Enter name: NUR ZAHRAH BT MALIK
Enter IC Number: 001215012345
Enter phone number: 012-734567
Enter your address: N0 44, JLN G, TMN L.
Enter your email address: zARAH@gmail.com


      -------------------- S P E C I A L   O F F E R ----------------------
```

So basically #customer2 will go through the *same process* as what #customer1 went through.

# EXTRA STEP : But, if the name is filled as *"ADMIN"* instead. The expected screen will display an loading animation before *displaying the daily report.*



```
                                Loading...

                        ██████████████████████████



                Welcome ADMIN! You have worked hard today.
                This the report of the day!1

        ******************************************************************
                             - DAILY REPORT ON SALES -
        ******************************************************************

        Total Sales                             : RM 139.20

        Total customers                         : 3

        Total phonecases sold                   : 7

        Total Screen Printings                  : 4

        Total Embosssing Printings              : 2

        Total Debossing Printings               : 1

        Total 3D Sublimation Printings          : 0

        Total Plastic Material                  : 4

        Total Thermo-polyurathane Material      : 2

        Total Carbon Fiber Material             : 1

        Highest Prints                          : Screen Printing

        Lowest Prints                           : 3D Sublimation Printing

        Highest Material                        : Plastic

        Lowest Material                         : Carbon Fiber

        ******************************************************************
                                  See you again!
        ******************************************************************
```

4.0 – CONCLUSION

In conclusion of this project is selling phoncases where the source code includes various ways of manipulating the input, counting the calculations and displaying the output. Throughout the project, it is stated that the source code contains selection control structures, repetition control structures and the usage of functions to complete a program.

Furthermore, the usage of array and some creativity artwork in the program was the result of hardwork and teamwork of the group members to apply something beyond the expected guideline.

Finally, we learnt how to cooperate to make this project work successfully. We also learn how hard it is to design a coding. We hope that our coding will be accepted and applied by the users and the customers. We also hope that our project can be improved to the next level in the near future.

5.0 – CD POCKET