CHAPTER 1

INTRODUCTION

1.1 PROBLEM DEFINITION

Manually maintaining the records for every product is an extremely time-consuming task that is also quite tough. Managing the details is not recommended because if any issues happen, such as missing records that were stored offline, several issues will arise and it would be impossible to recover the data.

The problem faced by the company is they do not have any systematic system to record and keep their stock data. It is difficult for the admin to record the stock data quickly and safely because they only keep it in the logbook and not properly organized.

1.2 OBJECTIVES

- Stock analysis is the tactical examination of your warehouse's requirements in light of sales projections for the market.
- This entails keeping track of your inventory, the inputs produced in your warehouse, and the anticipated needs to deal with seasonal phases where you might observe demand peaks and valleys.
- Because it has an impact on the foundation of the entire structure—the sales—stock analysis tasks are vital to the growth of a corporation. Any other aspect of strategy would be useless without dependable product stockpiles.
- Inventory management is performed for various operational duties in order to facilitate operations. It is employed for a number of crucial functions, including:

1. Accessibility of materials.
2. Better Customer Service Level.
Maintaining Minimal Wastage and Losses.
4. Keeping Adequate Stock.
5. Storage That Is Economical.
6. Inventories' Cost Value Can Be Reduced.
7. Improving Sales of Products

1.3 METHODOLOGY TO BE FOLLOWED

First the username and password is created, then declaration of structure where product name, quantity and price are initialized. User name and password is been created to login and manage the stock of items. Using linked list and Arrays in this project admin can view the number of stock items, add an item with the list of quantity and price, delete or modify the stock of items. This concept helps to manage the stock database in simpler way.

1.4 EXPECTED OUTCOMES

Effective reordering will result from a retailer stock analyzer. The availability, fastest-moving, and lowest-supply items will all be disclosed in accurate reports. Making a wise purchasing selection is aided by it.
It aids companies in keeping track of their stock levels so they can decide when and how much to create.

1.5 HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirements:

1. Intel Pentium Pro processor or later.
2. RAM 512MB or more.
3. Storage of 128GB or more

Software Requirements:

1. Dev C
2. C Language Version.

CHAPTER 2

DATA STRUCTURES

2.1 INTRODUCTION TO DATA STRUCTURES

Computer science has a subfield called data structures. The study of data structures enables you to comprehend how information is arranged and how data flow is controlled to boost program or process efficiency. The logical link between data components is structurally represented by a data structure. This indicates that a data structure groups data items depending on how they are related to one another.

A organized collection of variables that are variously related to one another is provided by a data structure. It serves as the foundation for a programming tool that depicts the link between data items and enables programmers to quickly analyses data.

Data structure can be classified into two categories:

- Primitive data structure
- Non-primitive data structure

2.2 LINKED LISTS

A linked list is a type of data structure where each data element has a link or pointer to the one after it. The data element can be added to or removed from a linked list at any location on a linear list. The data pieces need not be kept in successive positions in linked lists either. Each piece of data is given its own block of memory to be allocated.

As a result, a linked list is viewed as a chain of nodes, which are data items or records. Each node in the list has a pointer field and an information field. The pointer field holds the addresses of the following nodes in the list, whereas the information field provides the actual data.

2.3 SINGLY LINKED LISTS

- A singly-linked list is a group of nodes connected in a sequential manner.
- There are two fields on each node of the single linked list.
- Data field, value field, and info field: These fields each include information that is kept in the node.

The address of the subsequent node is contained in the link field or next field.

- The connection field links the nodes together.

• The last node's connection is NULL, indicating that the linked list has come to an end.
• First or head holds the address of the first node in the linked list.

2.4 DOUBLY LINKED LISTS

- A sophisticated data structure known as a "doubly linked list" (DLL) is an improved version of a basic linked list in which each node merely contains a reference to the next node.
- Reverse traversing is not feasible since we can only go through the components in one way. A doubly-linked list was used to overcome this issue since each node holds the addresses of the nodes before and after it, allowing for both forward and backward traversal of the list.
- As a result, each node in a doubly-linked list has three components: the node that holds the actual item, as well as pointers that provide the addresses of the nodes that come before and after it in the sequence. In this subject, we will discover

2.5 ARRAYS

An ordered grouping of data items is generally referred to as an array. A form of data structure called an array contains data components in close proximity to one another. array is thought to be linear.
Data structure for storing identical data type items. Consequently, a linear homogeneous data structure is another name for it.
When declaring an array, we may give each of its elements starting values by wrapping the values in brackets.

There are three types of arrays: one-dimensional [1D], two-dimensional [2D], and multidimensional arrays.

One-dimensional Array: There is only one row of items in it. The storage location is arranged in increasing order.

Two-dimensional Array: The data items are arranged in rows and columns. An alternative name for it is a matrix.

Multidimensional Array: The definition of an array of arrays is a multidimensional array. The number of indices or dimensions that a multidimensional array can have is not limited to two. The number of indexes they use is up to you.

CHAPTER 3

DESIGN

3.1 DESIGN GOALS

Our design mechanism includes two main things:

1. Track of stock (gadgets).
2. Edit to users desire and reduced time consumption.

3.2 ALGORITHM / PSEUDOCODE

```
1. start-Login
2. Enter username and password
3. Username: imManager, Password:grow123
 If correct username and password. If correct password then access granted else display login attempts failed.
4. Welcome To Gadget's Stock Analyzer.
5. Homescreen →Enter the choice of your department.
6. Welcome to for example: laptop database,
    • Add new laptop item to database.
    • Show all laptop items available.
    • Modify records.
    • Delete records.
    • Go to main menu.
7. Print "your choice:"
Case :
• Enter name of  new item :-
• Is available in quantity:-
• Enter the price:-
• Add another record (y/n):-
8. Display details.
9. Stop-Exit
```

3.3 FLOWCHART

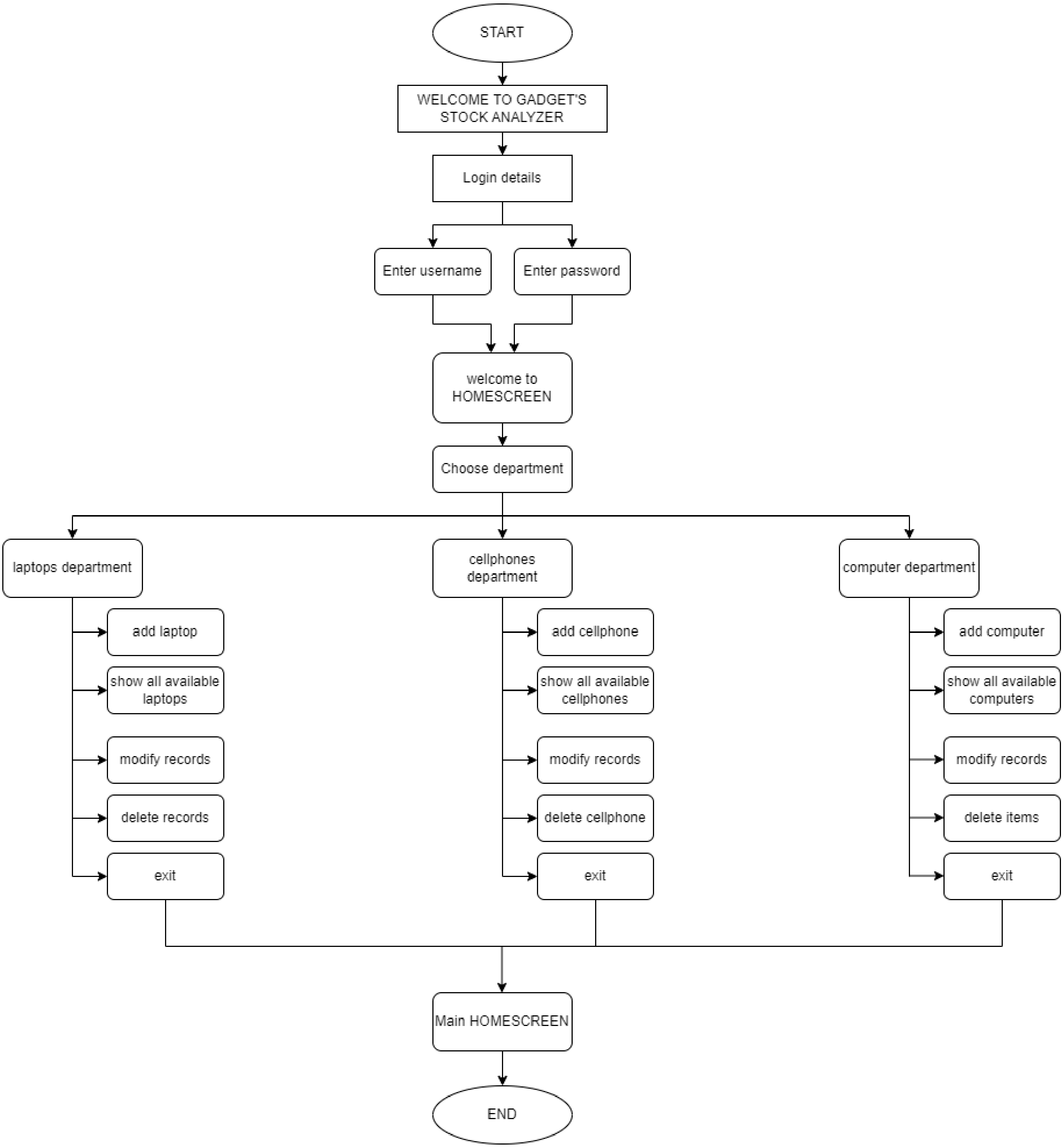
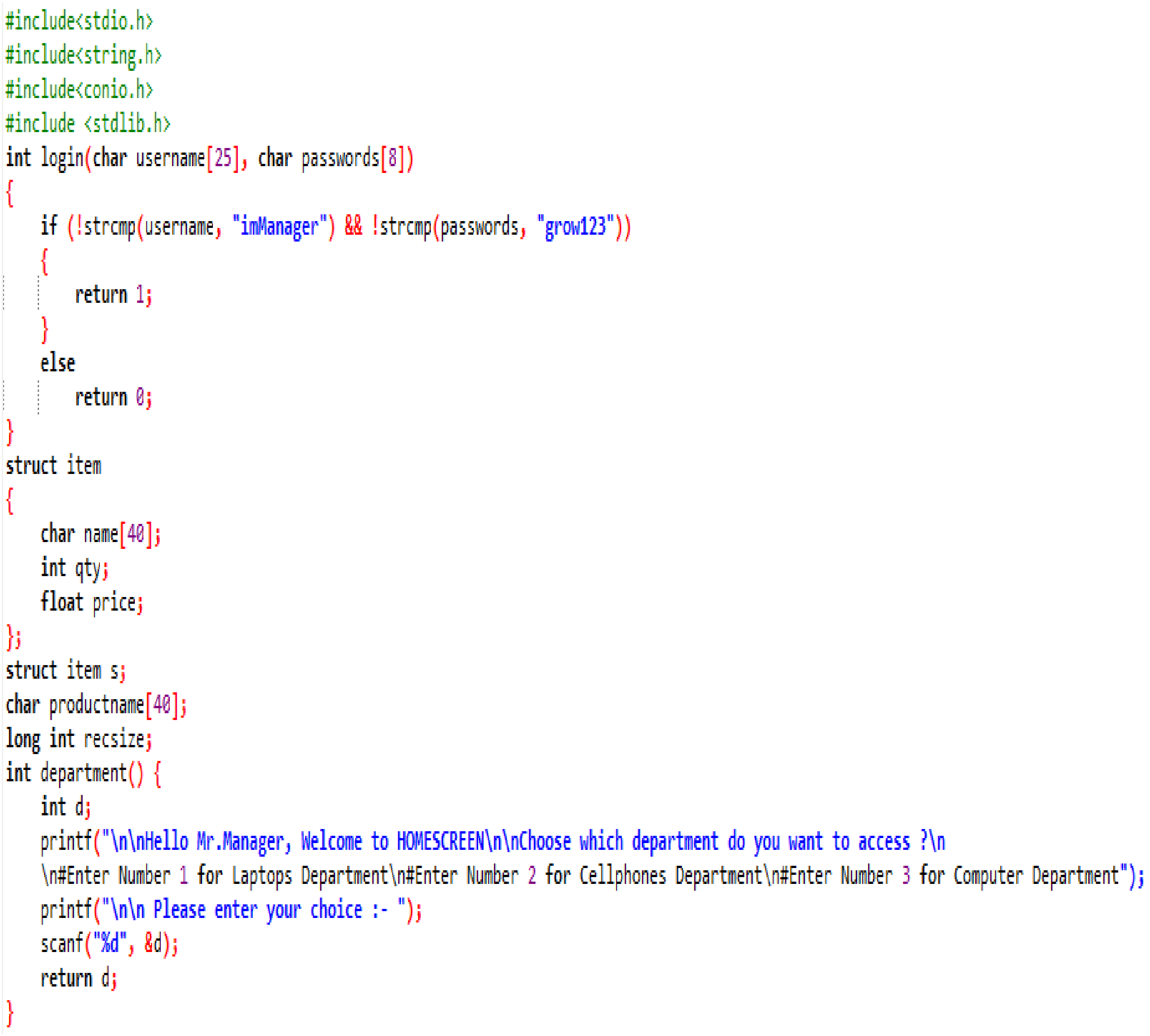


Fig 3.3: Design Flowchart

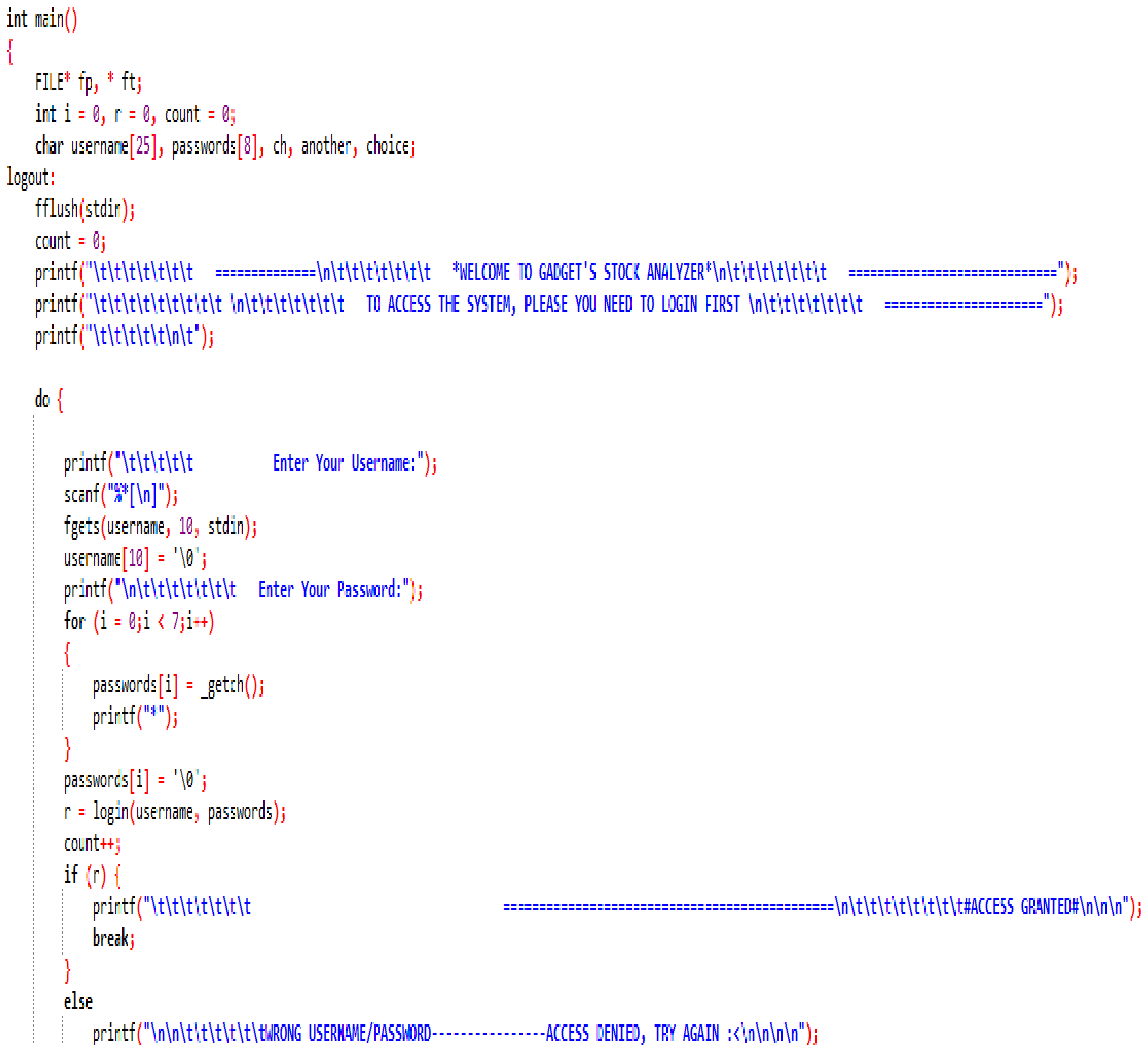
CHAPTER 4

IMPLEMENTATION

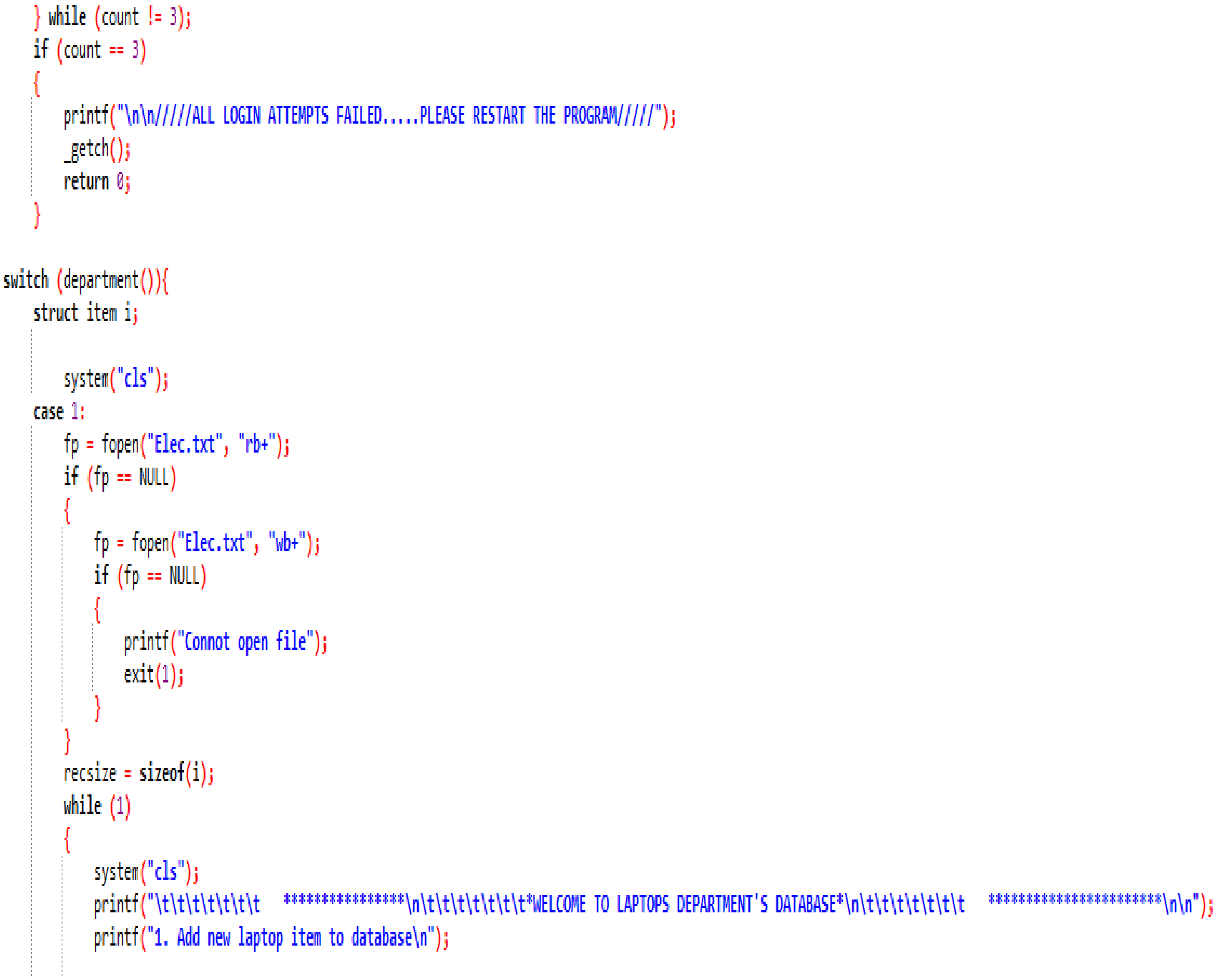
4.1 MODULE 1: Code



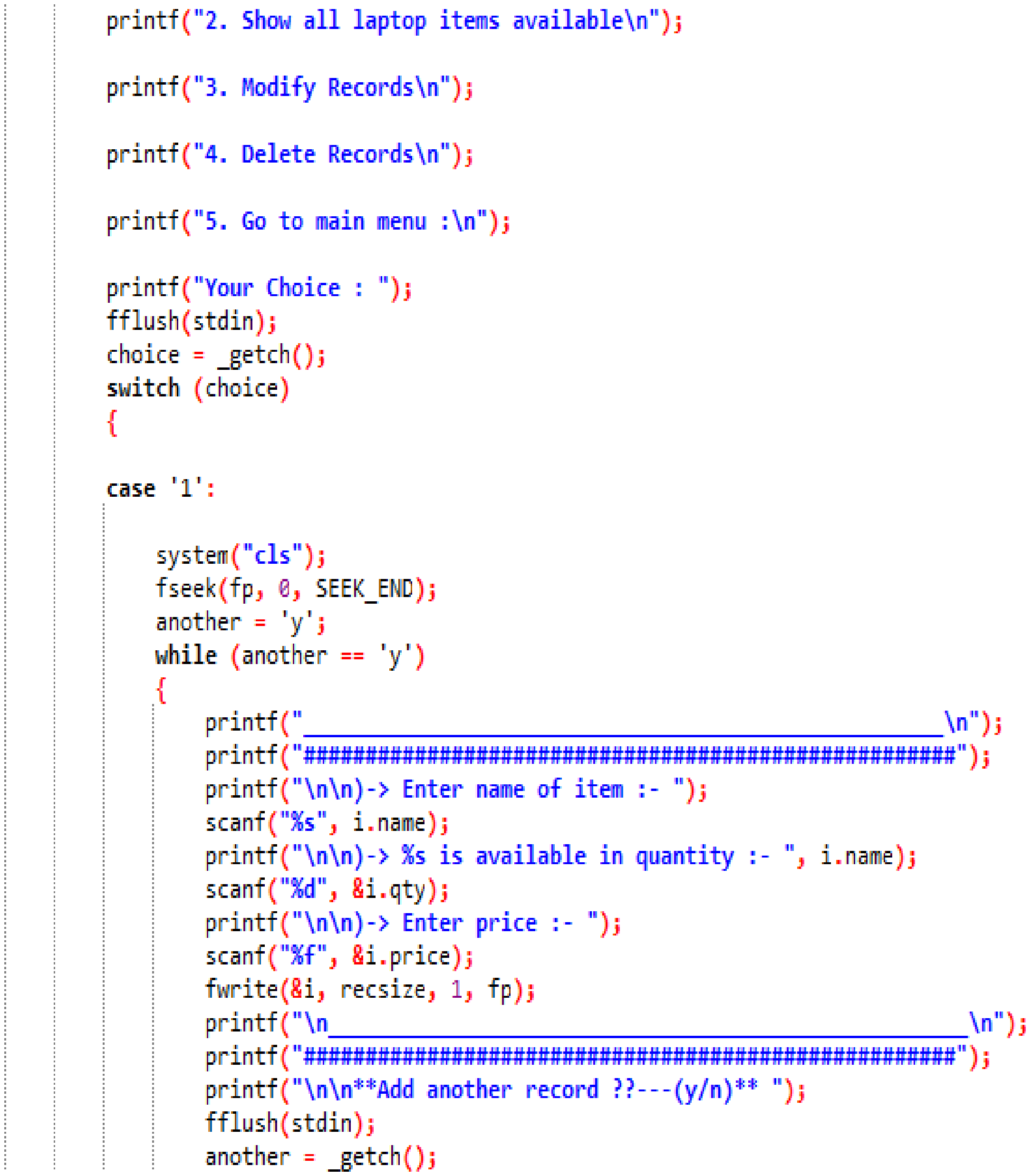
```
#include<stdio.h>
#include<string.h>
#include<conio.h>
#include <stdlib.h>
int login(char username[25], char passwords[8])
{
    if (!strcmp(username, "imManager") && !strcmp(passwords, "grow123"))
    {
        return 1;
    }
    else
        return 0;
}
struct item
{
    char name[40];
    int qty;
    float price;
};
struct item s;
char productname[40];
long int recsize;
int department() {
    int d;
    printf("\n\nHello Mr.Manager, Welcome to HOMESCREEN\n\nChoose which department do you want to access ?\n
    \n#Enter Number 1 for Laptops Department\n#Enter Number 2 for Cellphones Department\n#Enter Number 3 for Computer Department");
    printf("\n\n Please enter your choice :- ");
    scanf("%d", &d);
    return d;
}
```



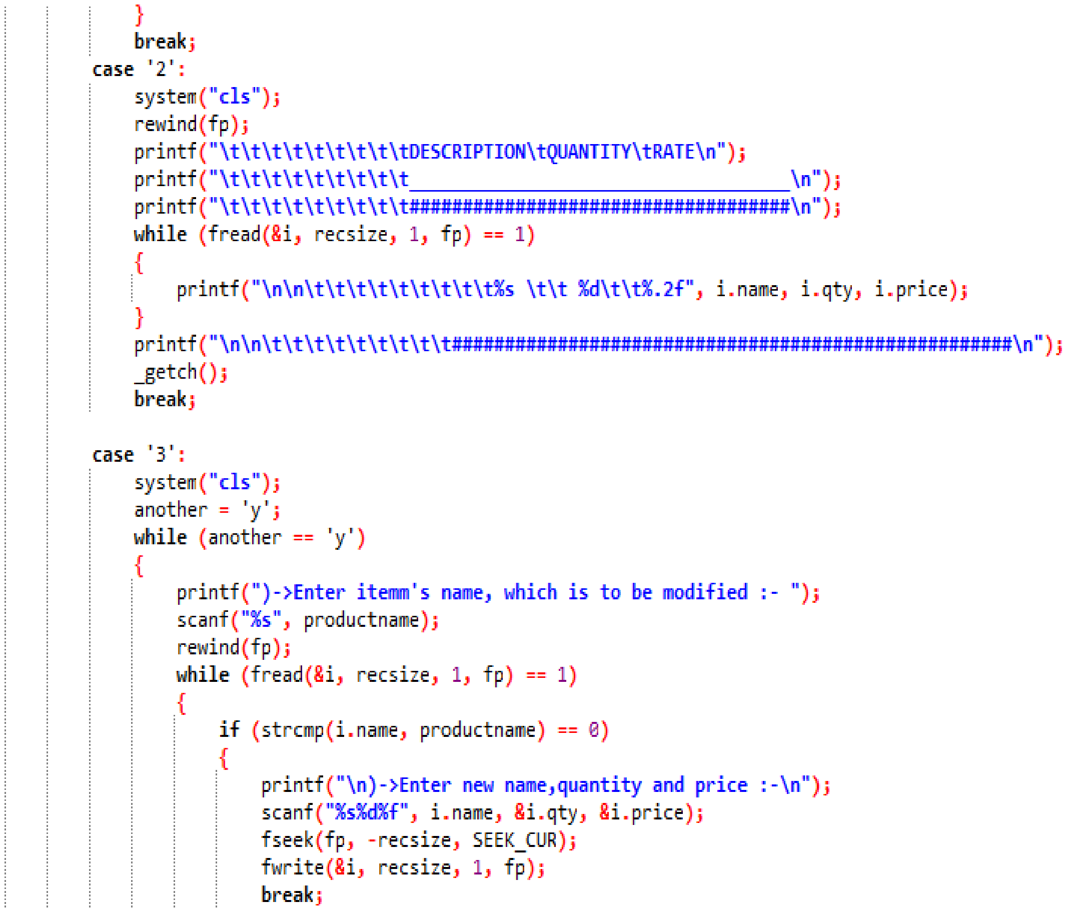
```c
int main()
{
    FILE* fp, * ft;
    int i = 0, r = 0, count = 0;
    char username[25], passwords[8], ch, another, choice;
logout:
    fflush(stdin);
    count = 0;
    printf("\t\t\t\t\t\t\t   ==============\n\t\t\t\t\t\t\t   *WELCOME TO GADGET'S STOCK ANALYZER*\n\t\t\t\t\t\t\t   ============================");
    printf("\t\t\t\t\t\t\t\t\t \n\t\t\t\t\t\t\t   TO ACCESS THE SYSTEM, PLEASE YOU NEED TO LOGIN FIRST \n\t\t\t\t\t\t\t   ======================");
    printf("\t\t\t\t\t\n\t");

    do {

        printf("\t\t\t\t\t          Enter Your Username:");
        scanf("%*[\n]");
        fgets(username, 10, stdin);
        username[10] = '\0';
        printf("\n\t\t\t\t\t\t\t   Enter Your Password:");
        for (i = 0;i < 7;i++)
        {
            passwords[i] = _getch();
            printf("*");
        }
        passwords[i] = '\0';
        r = login(username, passwords);
        count++;
        if (r) {
            printf("\t\t\t\t\t\t\t                              ============================================\n\t\t\t\t\t\t\t\t#ACCESS GRANTED#\n\n\n");
            break;
        }
        else
            printf("\n\n\t\t\t\t\t\t\tWRONG USERNAME/PASSWORD---------------ACCESS DENIED, TRY AGAIN :<\n\n\n\n");
```



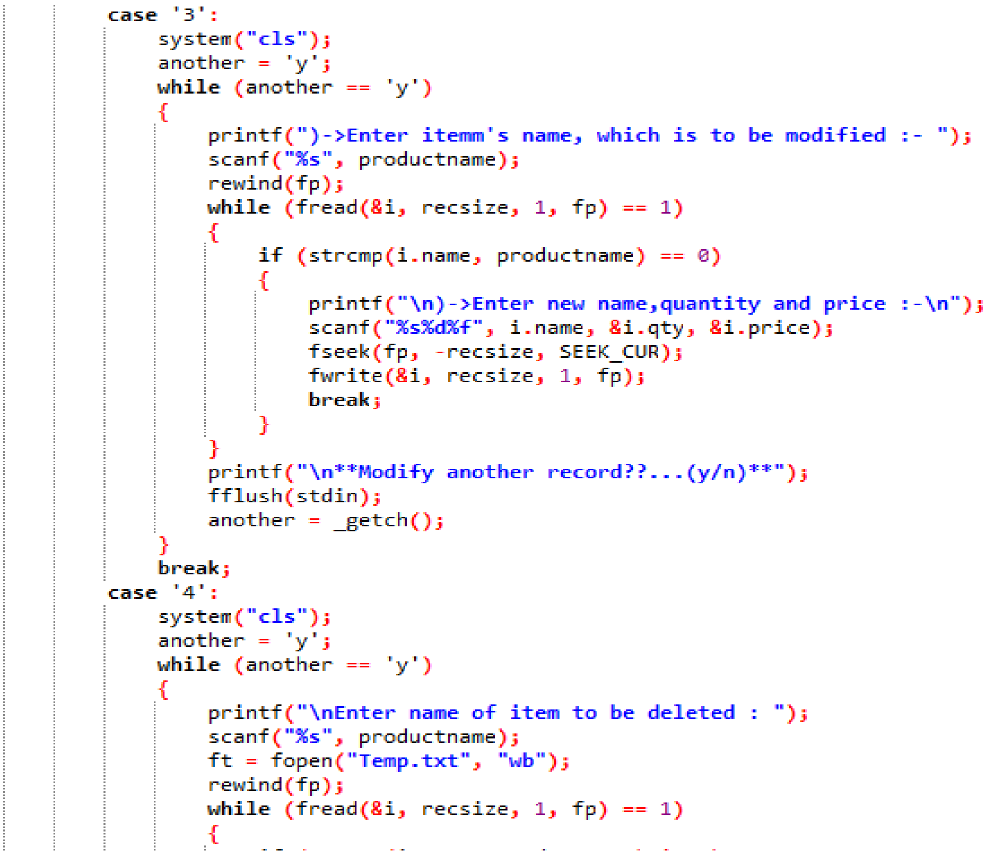
```
    } while (count != 3);
    if (count == 3)
    {
        printf("\n\n/////ALL LOGIN ATTEMPTS FAILED.....PLEASE RESTART THE PROGRAM/////");
        _getch();
        return 0;
    }

switch (department()){
    struct item i;

        system("cls");
    case 1:
        fp = fopen("Elec.txt", "rb+");
        if (fp == NULL)
        {
            fp = fopen("Elec.txt", "wb+");
            if (fp == NULL)
            {
                printf("Connot open file");
                exit(1);
            }
        }
        recsize = sizeof(i);
        while (1)
        {
            system("cls");
            printf("\t\t\t\t\t\t\t   ****************\n\t\t\t\t\t\t\t*WELCOME TO LAPTOPS DEPARTMENT'S DATABASE*\n\t\t\t\t\t\t\t   ***********************\n\n");
            printf("1. Add new laptop item to database\n");
```



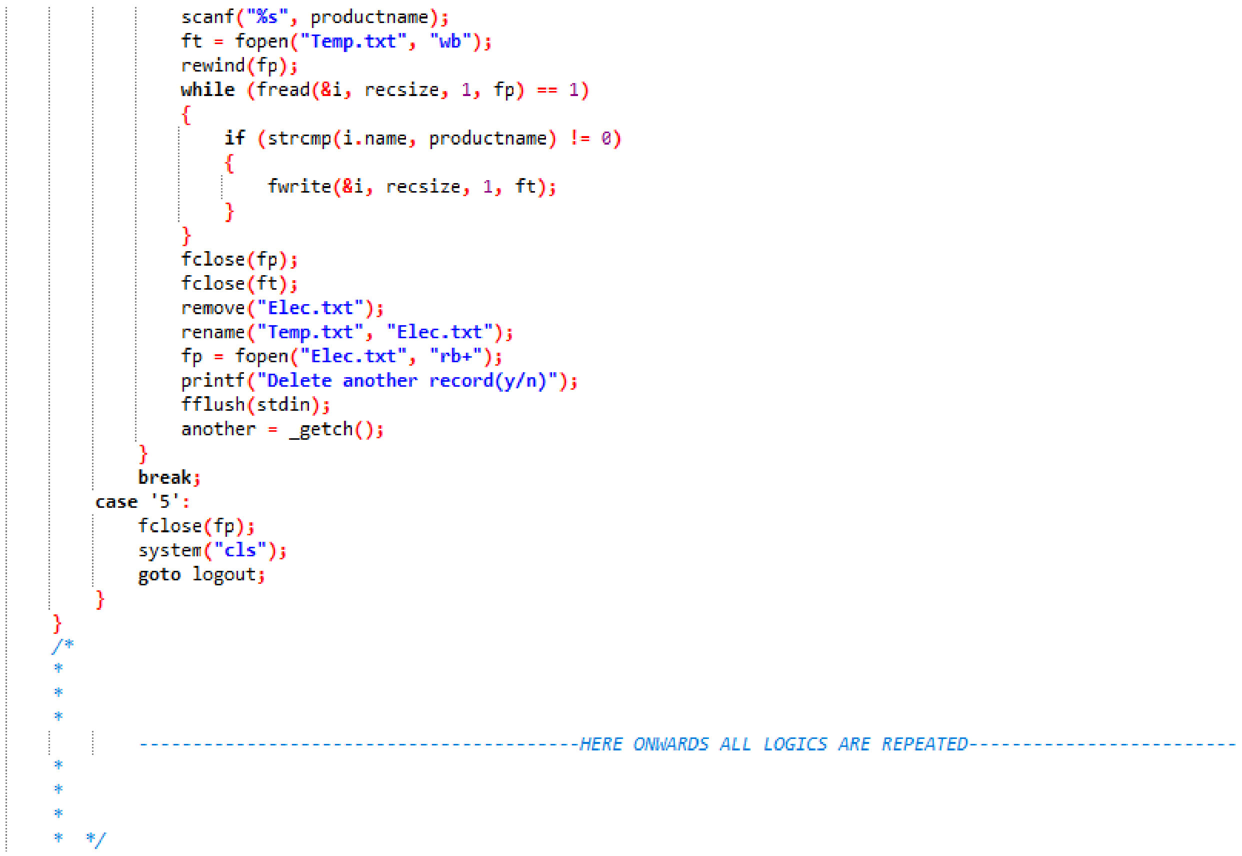
```
printf("2. Show all laptop items available\n");

printf("3. Modify Records\n");

printf("4. Delete Records\n");

printf("5. Go to main menu :\n");

printf("Your Choice : ");
fflush(stdin);
choice = _getch();
switch (choice)
{

case '1':

    system("cls");
    fseek(fp, 0, SEEK_END);
    another = 'y';
    while (another == 'y')
    {
        printf("____________________________________________________\n");
        printf("#####################################################");
        printf("\n\n)-> Enter name of item :- ");
        scanf("%s", i.name);
        printf("\n\n)-> %s is available in quantity :- ", i.name);
        scanf("%d", &i.qty);
        printf("\n\n)-> Enter price :- ");
        scanf("%f", &i.price);
        fwrite(&i, recsize, 1, fp);
        printf("\n____________________________________________________\n");
        printf("#####################################################");
        printf("\n\n**Add another record ??---(y/n)** ");
        fflush(stdin);
        another = _getch();
```



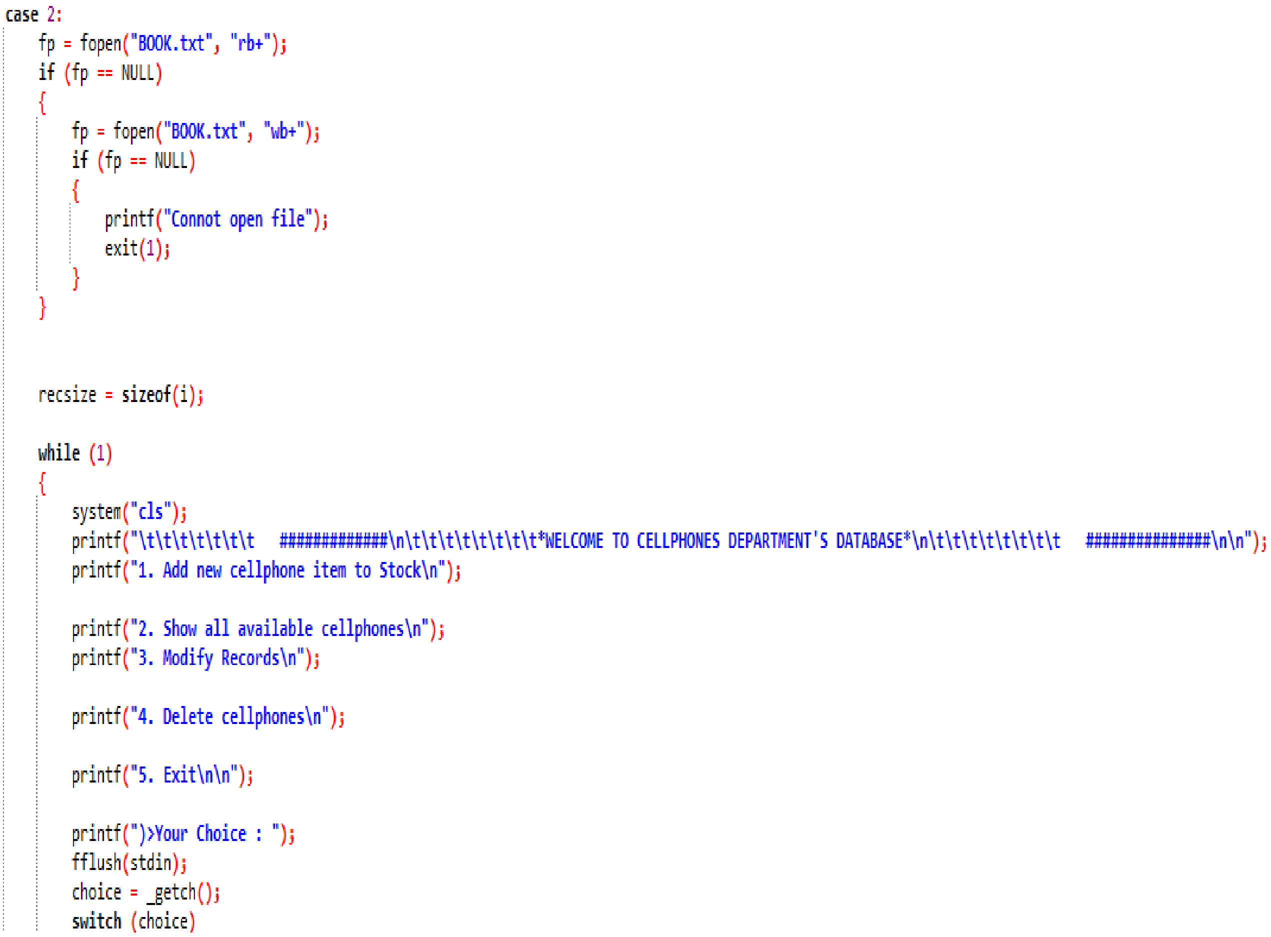
```
        }
        break;
    case '2':
        system("cls");
        rewind(fp);
        printf("\t\t\t\t\t\t\t\t\tDESCRIPTION\tQUANTITY\tRATE\n");
        printf("\t\t\t\t\t\t\t\t\t_____________________________________\n");
        printf("\t\t\t\t\t\t\t\t\t#####################################\n");
        while (fread(&i, recsize, 1, fp) == 1)
        {
            printf("\n\n\t\t\t\t\t\t\t\t\t\t%s \t\t %d\t\t%.2f", i.name, i.qty, i.price);
        }
        printf("\n\n\t\t\t\t\t\t\t\t\t\t########################################################\n");
        _getch();
        break;

    case '3':
        system("cls");
        another = 'y';
        while (another == 'y')
        {
            printf(")->Enter itemm's name, which is to be modified :- ");
            scanf("%s", productname);
            rewind(fp);
            while (fread(&i, recsize, 1, fp) == 1)
            {
                if (strcmp(i.name, productname) == 0)
                {
                    printf("\n)->Enter new name,quantity and price :-\n");
                    scanf("%s%d%f", i.name, &i.qty, &i.price);
                    fseek(fp, -recsize, SEEK_CUR);
                    fwrite(&i, recsize, 1, fp);
                    break;
```



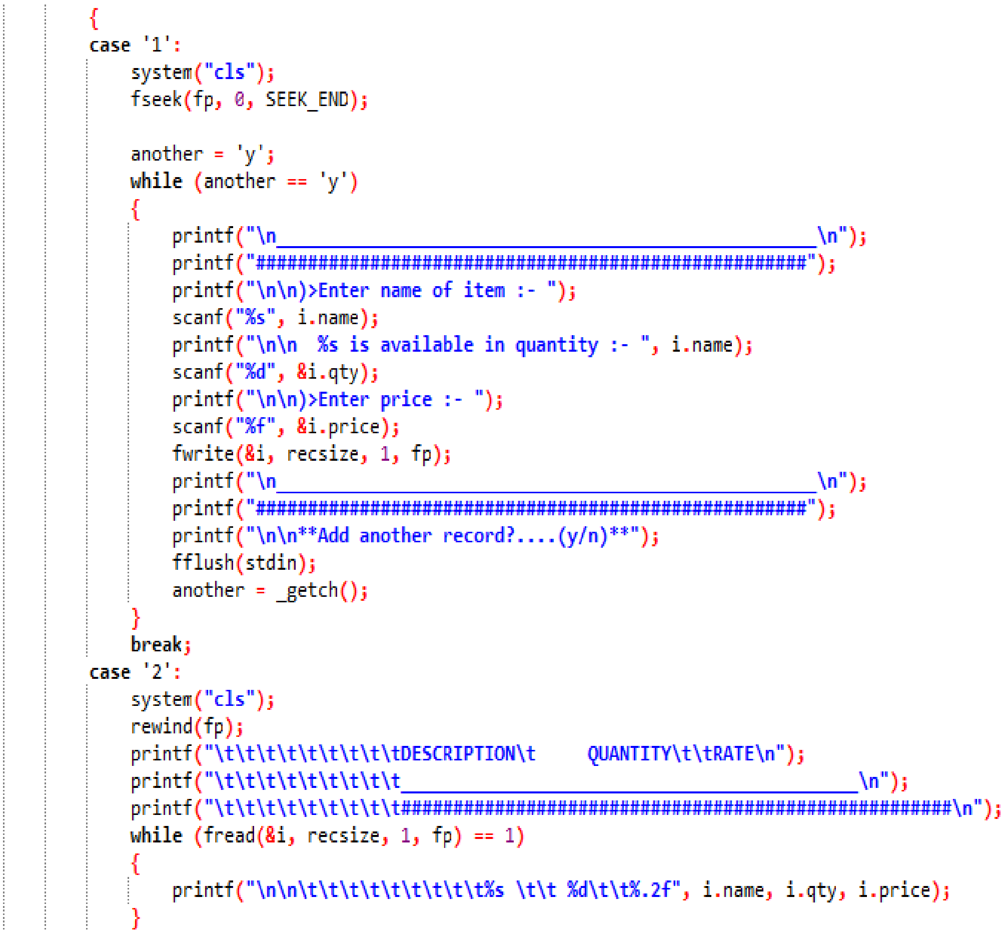
```
    case '3':
        system("cls");
        another = 'y';
        while (another == 'y')
        {
            printf(")->Enter itemm's name, which is to be modified :- ");
            scanf("%s", productname);
            rewind(fp);
            while (fread(&i, recsize, 1, fp) == 1)
            {
                if (strcmp(i.name, productname) == 0)
                {
                    printf("\n)->Enter new name,quantity and price :-\n");
                    scanf("%s%d%f", i.name, &i.qty, &i.price);
                    fseek(fp, -recsize, SEEK_CUR);
                    fwrite(&i, recsize, 1, fp);
                    break;
                }
            }
            printf("\n**Modify another record??...(y/n)**");
            fflush(stdin);
            another = _getch();
        }
        break;
    case '4':
        system("cls");
        another = 'y';
        while (another == 'y')
        {
            printf("\nEnter name of item to be deleted : ");
            scanf("%s", productname);
            ft = fopen("Temp.txt", "wb");
            rewind(fp);
            while (fread(&i, recsize, 1, fp) == 1)
            {
```



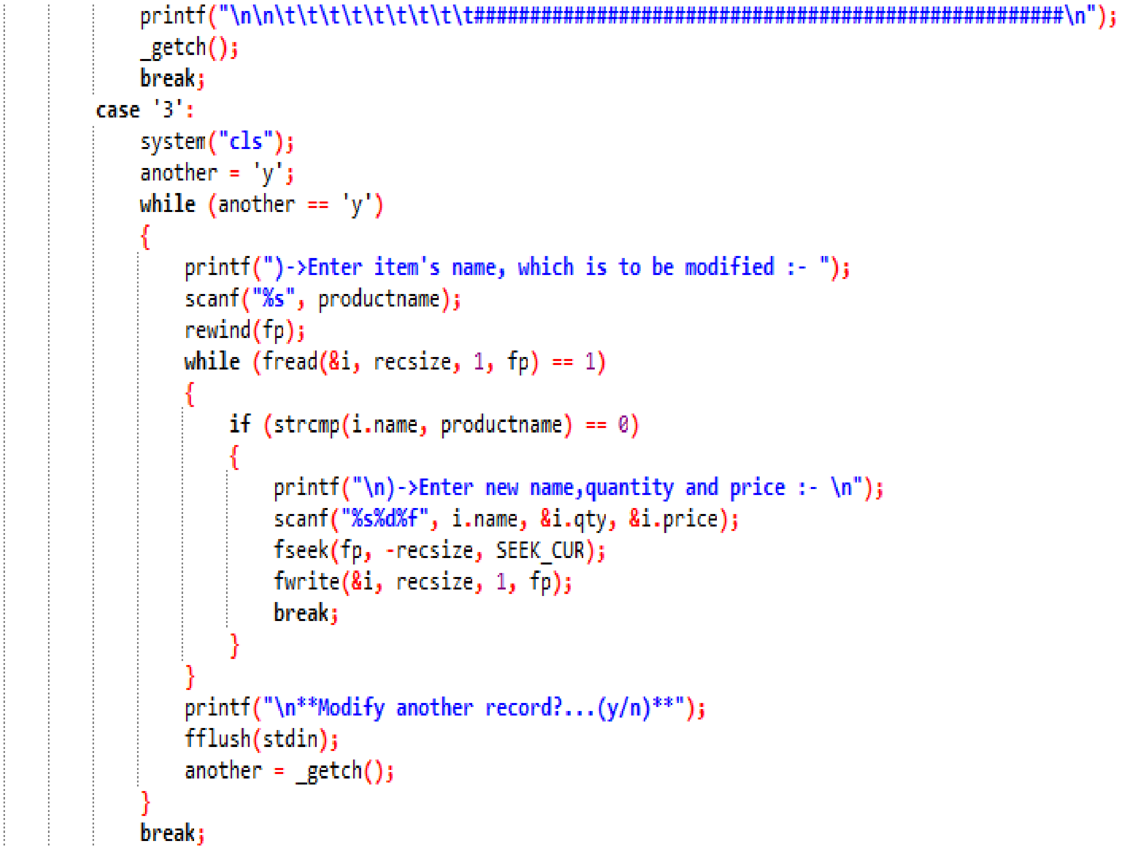
```
                scanf("%s", productname);
                ft = fopen("Temp.txt", "wb");
                rewind(fp);
                while (fread(&i, recsize, 1, fp) == 1)
                {
                    if (strcmp(i.name, productname) != 0)
                    {
                        fwrite(&i, recsize, 1, ft);
                    }
                }
                fclose(fp);
                fclose(ft);
                remove("Elec.txt");
                rename("Temp.txt", "Elec.txt");
                fp = fopen("Elec.txt", "rb+");
                printf("Delete another record(y/n)");
                fflush(stdin);
                another = _getch();
            }
            break;
        case '5':
            fclose(fp);
            system("cls");
            goto logout;
        }
    }
    /*
    *
    *
    *
            -----------------------------------------HERE ONWARDS ALL LOGICS ARE REPEATED-------------------------
    *
    *
    *
    *  */
```



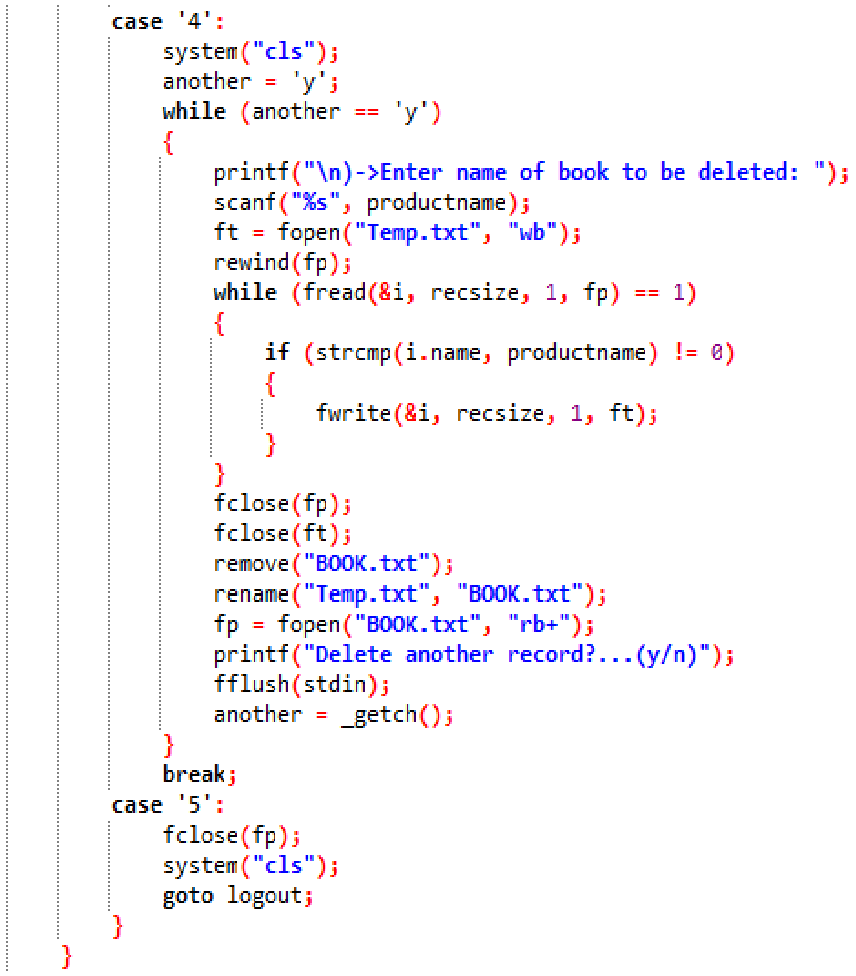
```
case 2:
    fp = fopen("BOOK.txt", "rb+");
    if (fp == NULL)
    {
        fp = fopen("BOOK.txt", "wb+");
        if (fp == NULL)
        {
            printf("Connot open file");
            exit(1);
        }
    }


    recsize = sizeof(i);

    while (1)
    {
        system("cls");
        printf("\t\t\t\t\t\t\t   #############\n\t\t\t\t\t\t\t\t*WELCOME TO CELLPHONES DEPARTMENT'S DATABASE*\n\t\t\t\t\t\t\t\t   ###############\n\n");
        printf("1. Add new cellphone item to Stock\n");

        printf("2. Show all available cellphones\n");
        printf("3. Modify Records\n");

        printf("4. Delete cellphones\n");

        printf("5. Exit\n\n");

        printf(")>Your Choice : ");
        fflush(stdin);
        choice = _getch();
        switch (choice)
```



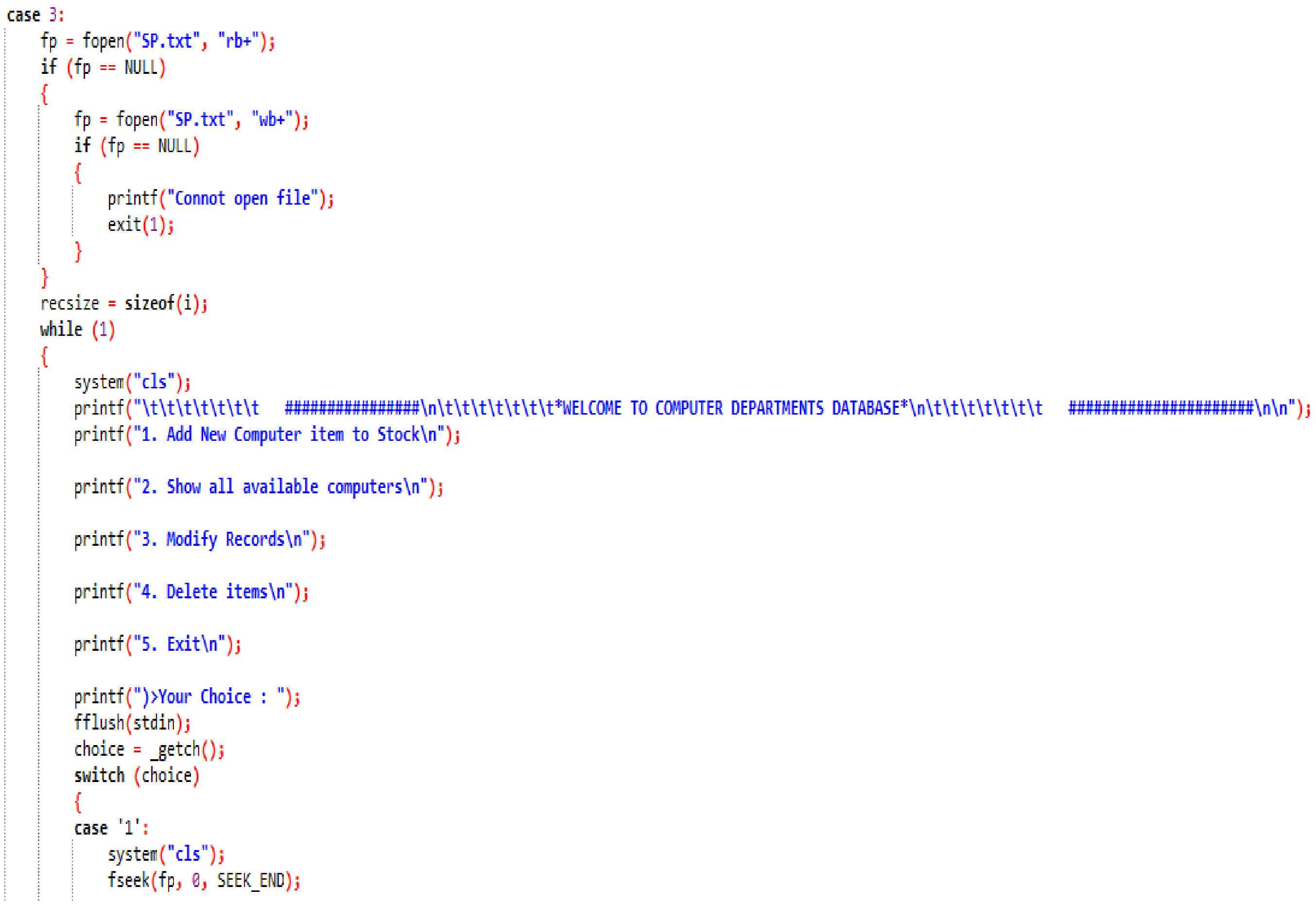
```
{
case '1':
    system("cls");
    fseek(fp, 0, SEEK_END);

    another = 'y';
    while (another == 'y')
    {
        printf("\n____________________________________________________\n");
        printf("#####################################################");
        printf("\n\n)>Enter name of item :- ");
        scanf("%s", i.name);
        printf("\n\n  %s is available in quantity :- ", i.name);
        scanf("%d", &i.qty);
        printf("\n\n)>Enter price :- ");
        scanf("%f", &i.price);
        fwrite(&i, recsize, 1, fp);
        printf("\n____________________________________________________\n");
        printf("#####################################################");
        printf("\n\n**Add another record?....(y/n)**");
        fflush(stdin);
        another = _getch();
    }
    break;
case '2':
    system("cls");
    rewind(fp);
    printf("\t\t\t\t\t\t\t\t\tDESCRIPTION\t     QUANTITY\t\tRATE\n");
    printf("\t\t\t\t\t\t\t\t\t____________________________________________\n");
    printf("\t\t\t\t\t\t\t\t\t######################################################\n");
    while (fread(&i, recsize, 1, fp) == 1)
    {
        printf("\n\n\t\t\t\t\t\t\t\t\t%s \t\t %d\t\t%.2f", i.name, i.qty, i.price);
    }
```



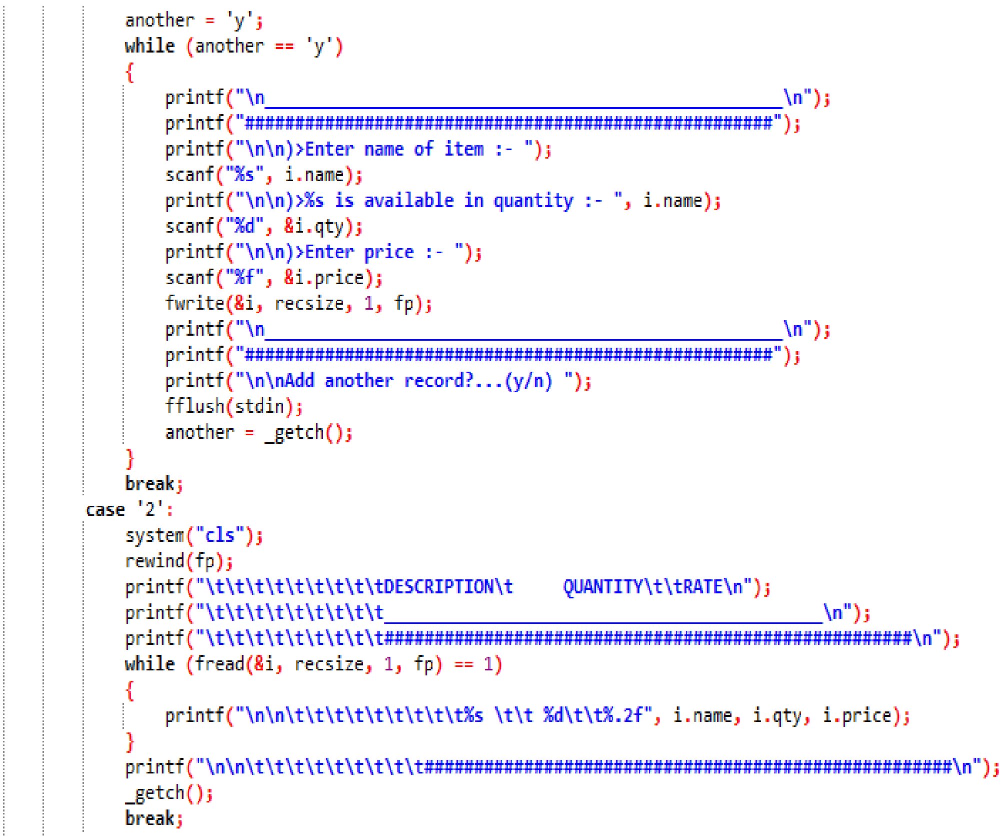
```
    printf("\n\n\t\t\t\t\t\t\t\t\t#####################################################\n");
    _getch();
    break;
case '3':
    system("cls");
    another = 'y';
    while (another == 'y')
    {
        printf(")->Enter item's name, which is to be modified :- ");
        scanf("%s", productname);
        rewind(fp);
        while (fread(&i, recsize, 1, fp) == 1)
        {
            if (strcmp(i.name, productname) == 0)
            {
                printf("\n)->Enter new name,quantity and price :- \n");
                scanf("%s%d%f", i.name, &i.qty, &i.price);
                fseek(fp, -recsize, SEEK_CUR);
                fwrite(&i, recsize, 1, fp);
                break;
            }
        }
        printf("\n**Modify another record?...(y/n)**");
        fflush(stdin);
        another = _getch();
    }
    break;
```



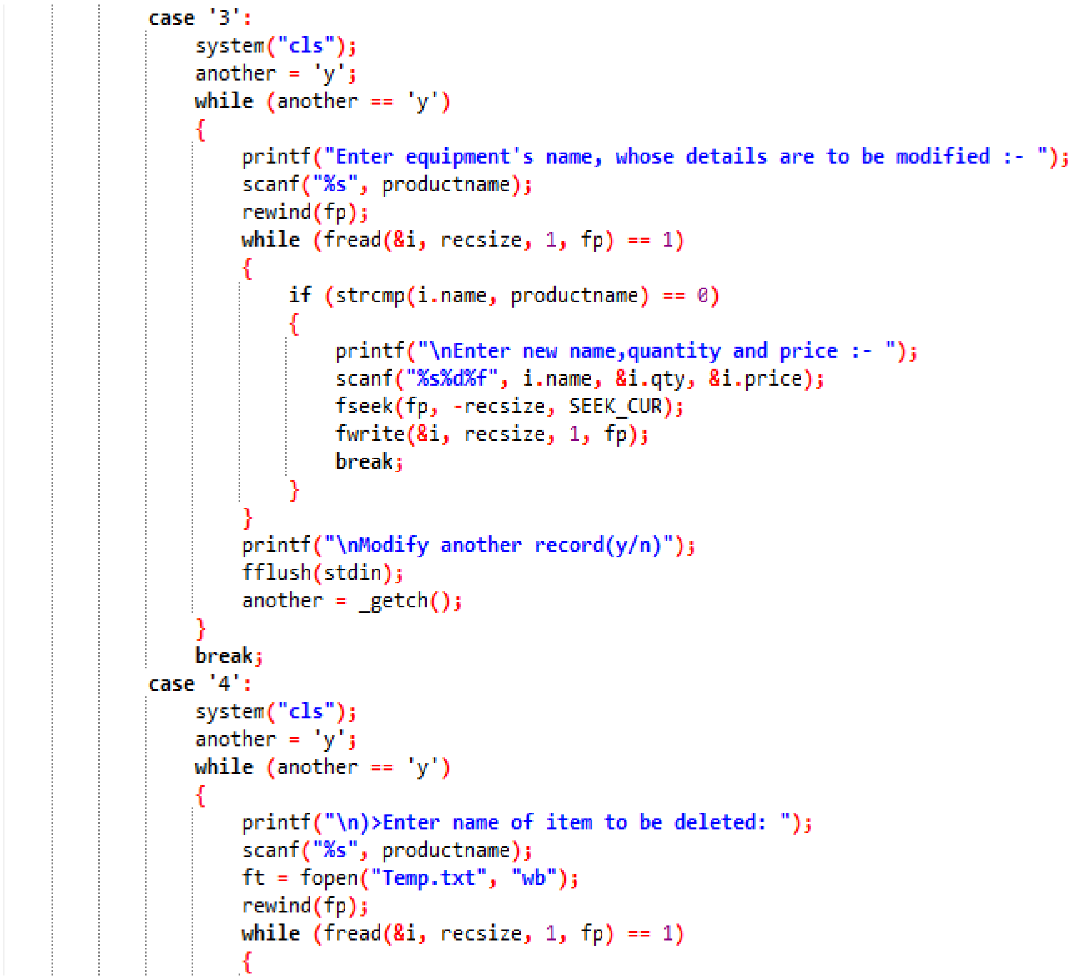
```
        case '4':
            system("cls");
            another = 'y';
            while (another == 'y')
            {
                printf("\n)->Enter name of book to be deleted: ");
                scanf("%s", productname);
                ft = fopen("Temp.txt", "wb");
                rewind(fp);
                while (fread(&i, recsize, 1, fp) == 1)
                {
                    if (strcmp(i.name, productname) != 0)
                    {
                        fwrite(&i, recsize, 1, ft);
                    }
                }
                fclose(fp);
                fclose(ft);
                remove("BOOK.txt");
                rename("Temp.txt", "BOOK.txt");
                fp = fopen("BOOK.txt", "rb+");
                printf("Delete another record?...(y/n)");
                fflush(stdin);
                another = _getch();
            }
            break;
        case '5':
            fclose(fp);
            system("cls");
            goto logout;
        }
    }
```



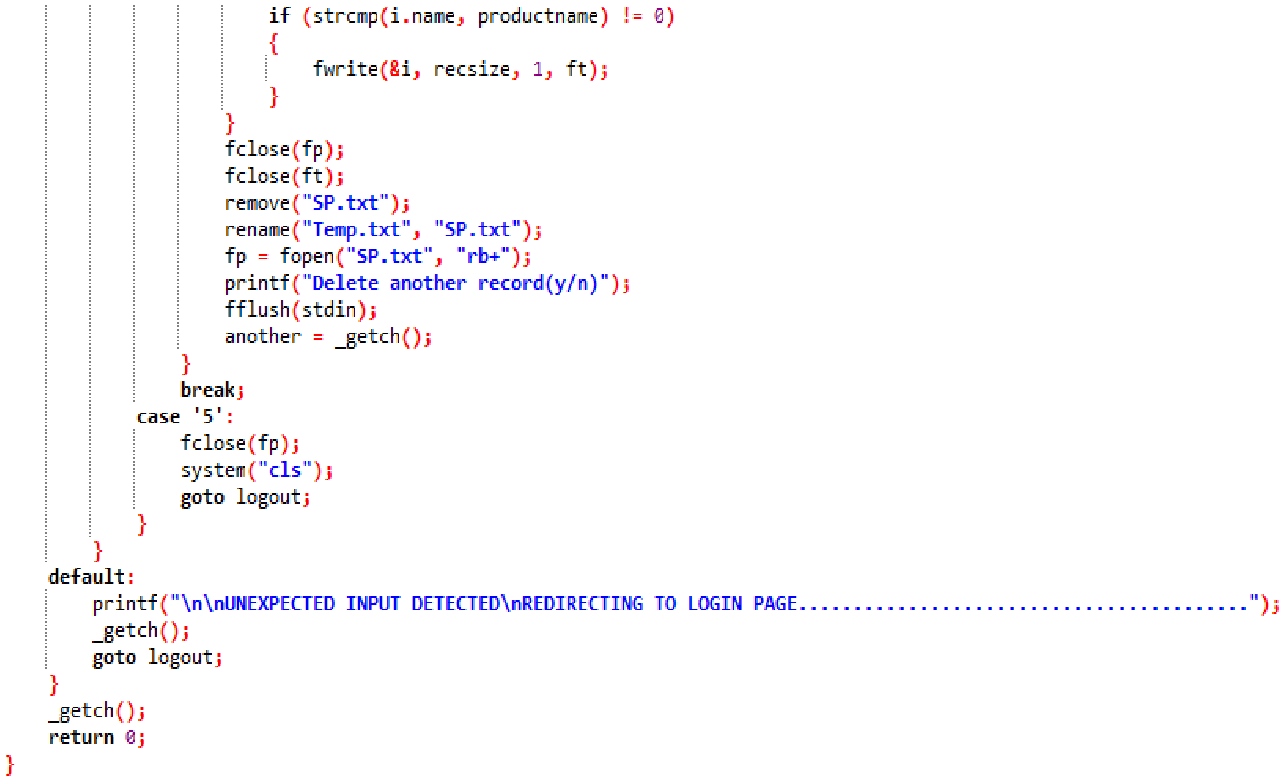
```
case 3:
    fp = fopen("SP.txt", "rb+");
    if (fp == NULL)
    {
        fp = fopen("SP.txt", "wb+");
        if (fp == NULL)
        {
            printf("Connot open file");
            exit(1);
        }
    }
    recsize = sizeof(i);
    while (1)
    {
        system("cls");
        printf("\t\t\t\t\t\t\t   ################\n\t\t\t\t\t\t\t*WELCOME TO COMPUTER DEPARTMENTS DATABASE*\n\t\t\t\t\t\t\t   #####################\n\n");
        printf("1. Add New Computer item to Stock\n");

        printf("2. Show all available computers\n");

        printf("3. Modify Records\n");

        printf("4. Delete items\n");

        printf("5. Exit\n");

        printf(")>Your Choice : ");
        fflush(stdin);
        choice = _getch();
        switch (choice)
        {
        case '1':
            system("cls");
            fseek(fp, 0, SEEK_END);
```



```
            another = 'y';
            while (another == 'y')
            {
                printf("\n_____________________________________________________\n");
                printf("######################################################");
                printf("\n\n)>Enter name of item :- ");
                scanf("%s", i.name);
                printf("\n\n)>%s is available in quantity :- ", i.name);
                scanf("%d", &i.qty);
                printf("\n\n)>Enter price :- ");
                scanf("%f", &i.price);
                fwrite(&i, recsize, 1, fp);
                printf("\n_____________________________________________________\n");
                printf("######################################################");
                printf("\n\nAdd another record?...(y/n) ");
                fflush(stdin);
                another = _getch();
            }
            break;
        case '2':
            system("cls");
            rewind(fp);
            printf("\t\t\t\t\t\t\t\t\tDESCRIPTION\t     QUANTITY\t\tRATE\n");
            printf("\t\t\t\t\t\t\t\t\t_____________________________________________\n");
            printf("\t\t\t\t\t\t\t\t\t#######################################################\n");
            while (fread(&i, recsize, 1, fp) == 1)
            {
                printf("\n\n\t\t\t\t\t\t\t\t\t%s \t\t %d\t\t%.2f", i.name, i.qty, i.price);
            }
            printf("\n\n\t\t\t\t\t\t\t\t\t#######################################################\n");
            _getch();
            break;
```



```
            case '3':
                system("cls");
                another = 'y';
                while (another == 'y')
                {
                    printf("Enter equipment's name, whose details are to be modified :- ");
                    scanf("%s", productname);
                    rewind(fp);
                    while (fread(&i, recsize, 1, fp) == 1)
                    {
                        if (strcmp(i.name, productname) == 0)
                        {
                            printf("\nEnter new name,quantity and price :- ");
                            scanf("%s%d%f", i.name, &i.qty, &i.price);
                            fseek(fp, -recsize, SEEK_CUR);
                            fwrite(&i, recsize, 1, fp);
                            break;
                        }
                    }
                    printf("\nModify another record(y/n)");
                    fflush(stdin);
                    another = _getch();
                }
                break;
            case '4':
                system("cls");
                another = 'y';
                while (another == 'y')
                {
                    printf("\n)>Enter name of item to be deleted: ");
                    scanf("%s", productname);
                    ft = fopen("Temp.txt", "wb");
                    rewind(fp);
                    while (fread(&i, recsize, 1, fp) == 1)
                    {
```

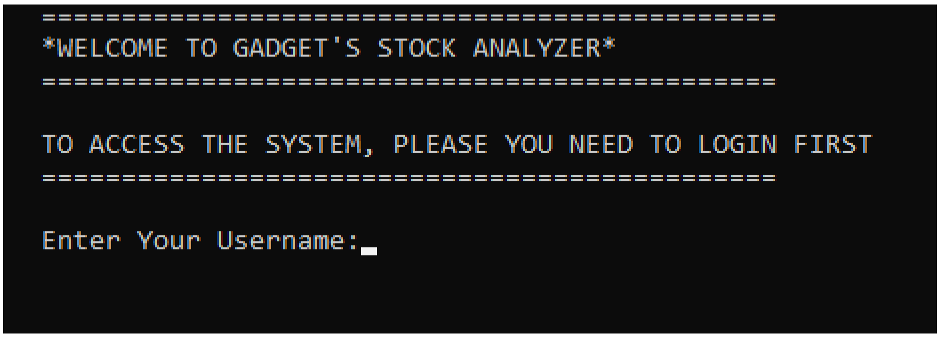


```
                        if (strcmp(i.name, productname) != 0)
                        {
                            fwrite(&i, recsize, 1, ft);
                        }
                    }
                    fclose(fp);
                    fclose(ft);
                    remove("SP.txt");
                    rename("Temp.txt", "SP.txt");
                    fp = fopen("SP.txt", "rb+");
                    printf("Delete another record(y/n)");
                    fflush(stdin);
                    another = _getch();
                }
                break;
            case '5':
                fclose(fp);
                system("cls");
                goto logout;
            }
        }
    default:
        printf("\n\nUNEXPECTED INPUT DETECTED\nREDIRECTING TO LOGIN PAGE.......................................");
        _getch();
        goto logout;
    }
    _getch();
    return 0;
}
```

CHAPTER 5

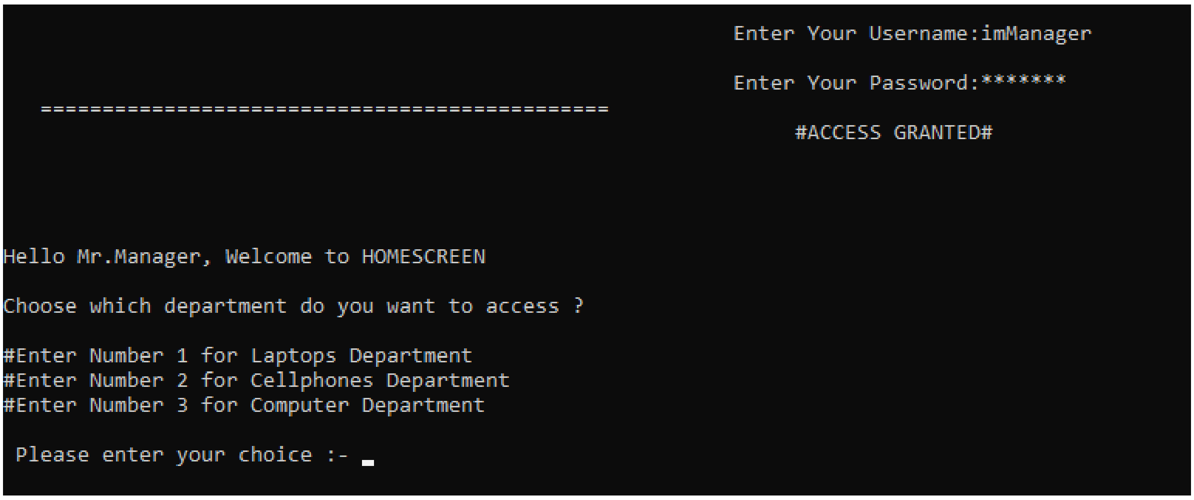
RESULTS

5.1 Snapshots



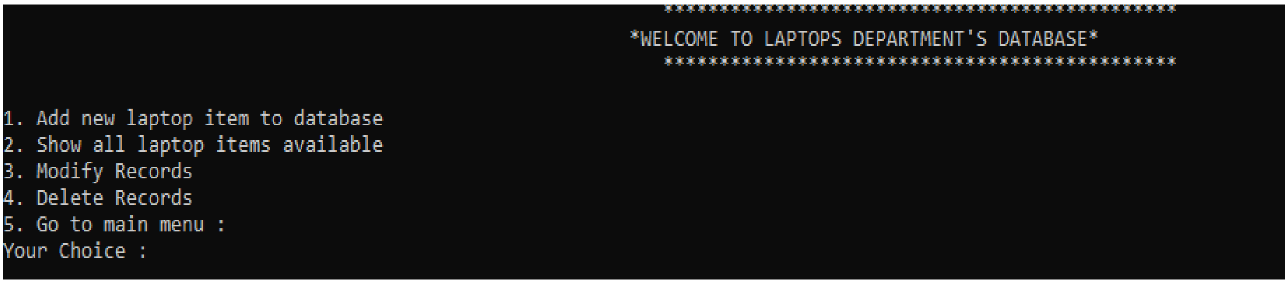
```
  ===============================================
  *WELCOME TO GADGET'S STOCK ANALYZER*
  ===============================================

  TO ACCESS THE SYSTEM, PLEASE YOU NEED TO LOGIN FIRST
  ===============================================

  Enter Your Username:_
```

Here is the output of the gadget stock analyzer. First it asks to enter the username and password details.



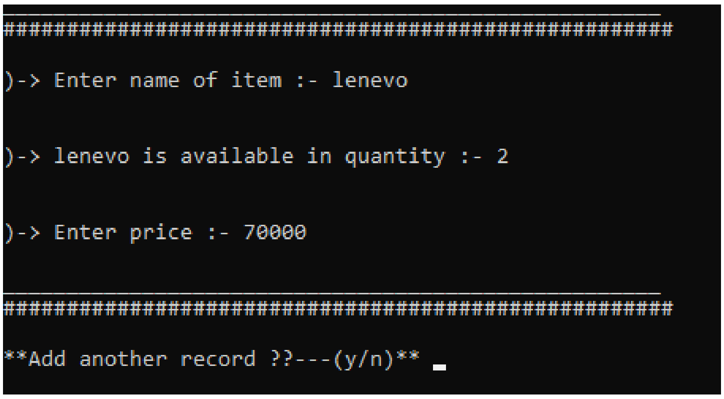
```
                                                            Enter Your Username:imManager

                                                            Enter Your Password:*******
   ==============================================
                                                                 #ACCESS GRANTED#




Hello Mr.Manager, Welcome to HOMESCREEN

Choose which department do you want to access ?

#Enter Number 1 for Laptops Department
#Enter Number 2 for Cellphones Department
#Enter Number 3 for Computer Department

 Please enter your choice :- _
```

Once you enter the username and password, an interface appears showing the above results.
It asks to choose a department from the three departments shown.

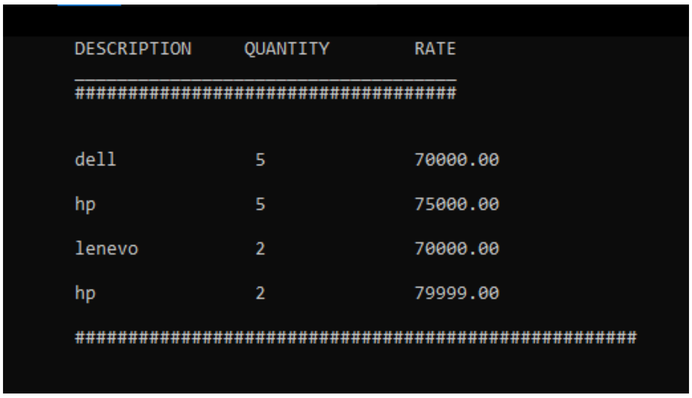


```
                                                                     ***********************************************
                                                                *WELCOME TO LAPTOPS DEPARTMENT'S DATABASE*
                                                                     ***********************************************


1. Add new laptop item to database
2. Show all laptop items available
3. Modify Records
4. Delete Records
5. Go to main menu :
Your Choice :
```

When you choose the first department, the following options appear under the first department.



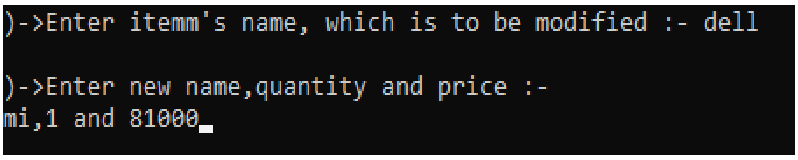
```
____________________________________________________________
############################################################

)-> Enter name of item :- lenevo


)-> lenevo is available in quantity :- 2


)-> Enter price :- 70000

____________________________________________________________
############################################################

**Add another record ??---(y/n)** _
```



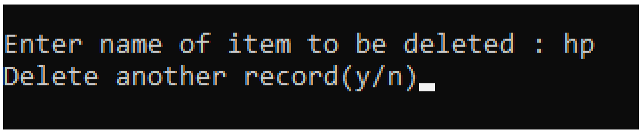
```
DESCRIPTION     QUANTITY        RATE
__________________________________
##################################


dell             5              70000.00

hp               5              75000.00

lenevo           2              70000.00

hp               2              79999.00

##################################################
```

option 1: The following results appear where you enter item, quantity and the price. You can add more records by choosing ‘y’ after you finish entering the current one.

option 2: it displays the output of all the registered items in the list.



```
)->Enter itemm's name, which is to be modified :- dell

)->Enter new name,quantity and price :-
mi,1 and 81000_
```

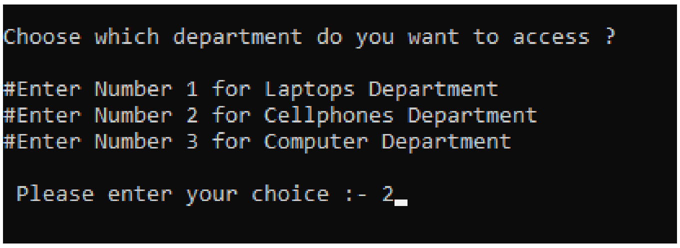


```
Enter name of item to be deleted : hp
Delete another record(y/n)_
```

option 3: We can modify the records by entering the new name, quantity and the price.

option 4: It helps In deleting item to be deleted. You can delete more records by choosing ‘y’ after you finish the current one.

option 5: You will get back to the main menu. You will exit the current interface.



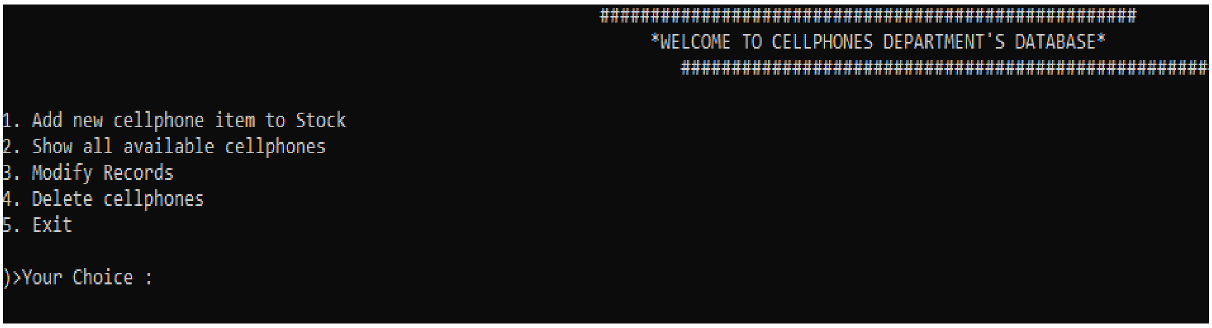
```
Choose which department do you want to access ?

#Enter Number 1 for Laptops Department
#Enter Number 2 for Cellphones Department
#Enter Number 3 for Computer Department

 Please enter your choice :- 2_
```

Here again we have the main menu interface, by choosing option 2.

By choosing option 2, we get the following options as above results.
The results have the same method as explained previously.

option 1- used to add a new cellphone to the stock record.
option 2- used to display all the available cellphones in the recorded list.

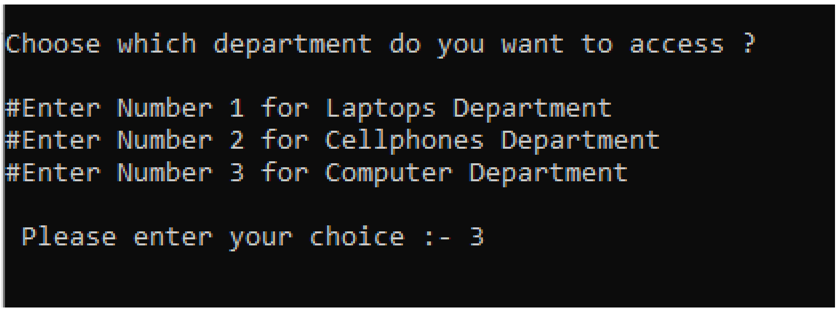
option 3- used to modify the recorded list if any modification is required.



```
                                                                    #####################################################
                                                                         *WELCOME TO CELLPHONES DEPARTMENT'S DATABASE*
                                                                            ####################################################


1. Add new cellphone item to Stock
2. Show all available cellphones
3. Modify Records
4. Delete cellphones
5. Exit

)>Your Choice :
```

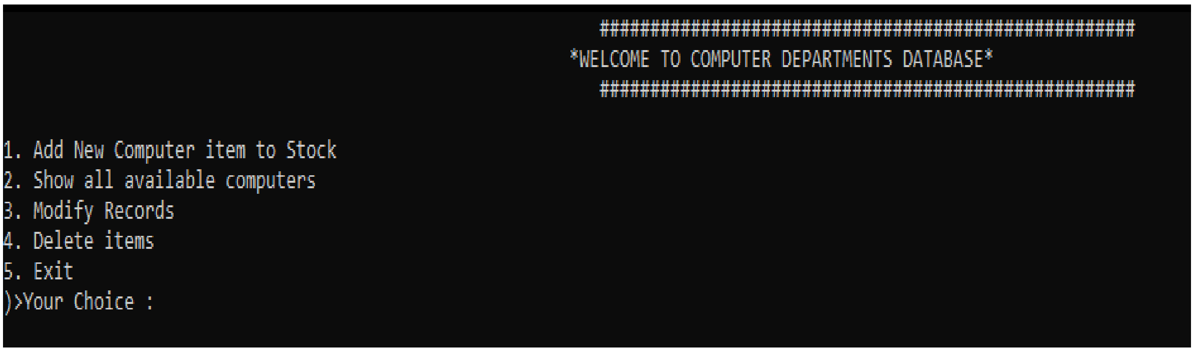
option 4- used to delete any record or any item if necessary.

option 5- used to exit or quit the interface and return to the main menu.



```
Choose which department do you want to access ?

#Enter Number 1 for Laptops Department
#Enter Number 2 for Cellphones Department
#Enter Number 3 for Computer Department

 Please enter your choice :- 3
```

Here again we come back to the main menu interface. By choosing option 3, we get the following results.



```
                                                           #####################################################
                                                        *WELCOME TO COMPUTER DEPARTMENTS DATABASE*
                                                           #####################################################


1. Add New Computer item to Stock
2. Show all available computers
3. Modify Records
4. Delete items
5. Exit
)>Your Choice :
```

By choosing option 3, we get the following options as above results.
The results have the same method as explained previously for the other 2 options.
Option 1 – used to add a new computer to the stock record.
Option 2 – used to display all the available computers in the recorded list.
Option 3 – used to modify the recorded list if any modification is required.
Option 4 – used to delete any record or any item is necessary.
Option 5 – used to exit or quit the interface and return to the main menu.

CHAPTER 6

CONCLUSION

The aims and design goals that the mini project had outlined in this report's objectives and design sections had been effectively attained.

As you can see, a stock analyzer is crucial to every firm and one of its most crucial components. If you can meet client demand in this area of the business depends on whether you are certain that you have all the supplies necessary to produce the desired result.

Planning, organizing, leading, and managing are the four managerial roles that enable this innovative problem-solving. The intended outcome is achieved when an organization uses its resources in a way that furthers its goals and mission. These might include improved production or efficiency, cost savings, error-proofing, or real-time inventory visibility. It's true that prices fluctuate based on your demands and the solution.

REFERENCES

[1] C programming by Dennis Richie.

[2] www.geeksforgeeks.org

[3] www.scribd.com

[4] dev c++