Supermarket Management System

Abstract

This mini-project “Supermarket management system project in C” is a console application using the C programming language. This project compiled in Code Blocks with the GCC compiler. In this console application, you can do basic supermarket management task like adding the items, view the added items, search the items, pay for the items, delete items, etc.

This application is based on file handling in C, where I have used a file-related function like fopen, fread, fwrite, etc. Another feature is that this project is protected by a password, so, only authorized people can login to the program.

Also to increase the readability, I have broken the application in different function. Each function of the project extensively use the file handing function, so it is also a great project to understand file handling.

Also this project uses structs as a data model to encapsulate the items properties.

Introduction

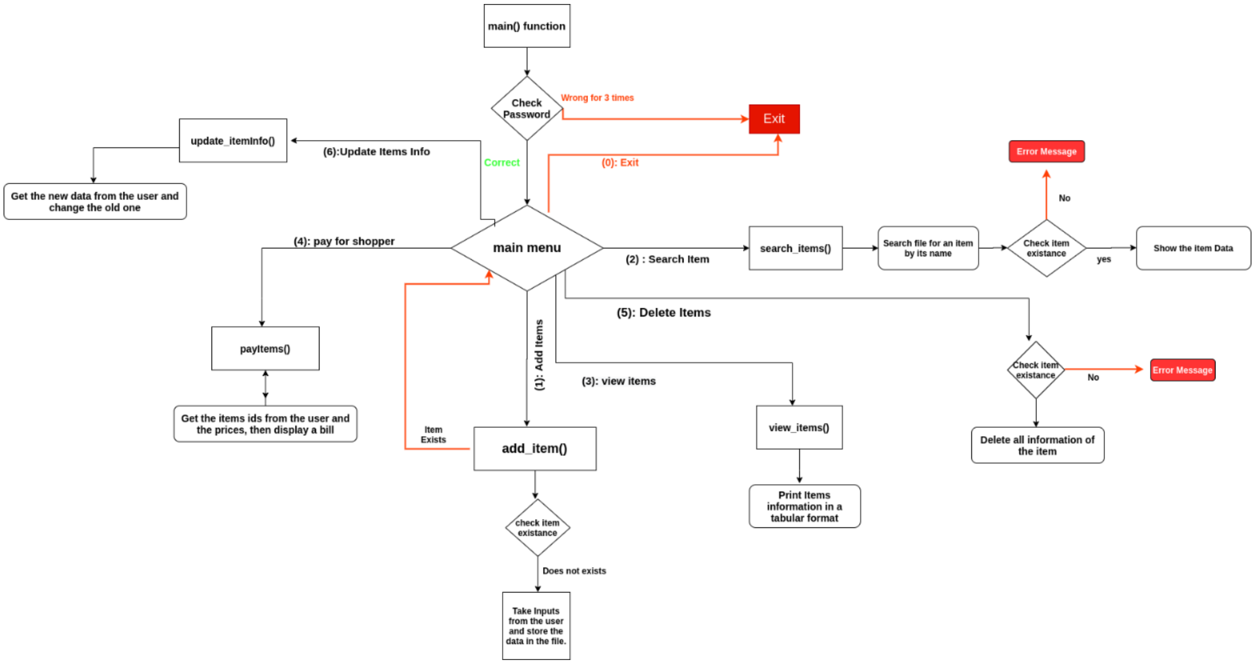
The program is mainly divided into 18 functions. Six of them are helper functions to facilitate the job of the main functions.

In this project, we need some basic functionality to exist like: adding an item, deleting an item, updating item, paying for this item, view all items, and update login password.

Every functionality has its own functions, it may need only one function to perform its job or it need many functions for that.

We will take snippets from the code and start to explain it intensively.

Flowchart



Implementation and Source Code

This portion of code shows the different libraries that are used in the program as well as the macros.

Line 8 defines a macro for the maximum size of the user name string as well as the password in line 9.

Line 10 defines the file name macro that will be used in overall the program to open file to get or manipulate data.

Line 12 defines a maximum size for the item name.

Line 14 defines a macro of the size of the struct of the credentials.

```
1.  #include <stdio.h>
2.  #include <time.h>
3.  #include <string.h>
4.  #include <stdlib.h>
5.  #include <ctype.h>
6.
7.
8.  #define MAX_SIZE_USER_NAME 30
9.  #define MAX_SIZE_PASSWORD  20
10. #define FILE_NAME  "Supermarket.bin"
11.
12. #define MAX_ITEM_NAME   50
13.
14. #define FILE_HEADER_SIZE  sizeof(sFileHeader)
```

These lines are used to define structs that act like the data model for the program. The sFileHeader struct used to encapsulate the credentials members like username and password. But, the s_itemInfo struct is used to encapsulate the different properties of the item like id, name, quantity, and price.

```
// Struct for login credentials
typedef struct
{
    char username[MAX_SIZE_USER_NAME];
    char password[MAX_SIZE_PASSWORD];
} sFileHeader;
typedef struct // to call in program
{
```

```c
    unsigned int item_id; // declare the integer data type
    char item_name[MAX_ITEM_NAME];// declare the character data type
    double amount;
    double price;

} s_itemInfo;
```

This is a helper function used to print any message passed to it in the center of the console.

```c
// This function prints the message in the middle of the head massage.
void printMessageCenter(const char* message)
{
    int len =0;
    int pos = 0;
    //calculate how many space need to print
    len = (78 - strlen(message))/2;
    printf("\t\t\t");
    for(pos =0 ; pos < len ; pos++)
    {
        //print space
        printf(" ");
    }
    //print message
    printf("%s",message);
}
```

This function is used to print the header message to the user based on the operation. If the operation for example is Add operation, then it will print ADD ITEMS under the name of the program.

```c
// It prints the message on the top of the console and prints the message for specific operation.
void headMessage(const char *message)
{
    system("cls");
    printf("\t\t\t###########################################################################");
    printf("\n\t\t\t############                                                   ############");
    printf("\n\t\t\t############      Supermarket management System Project in C   ############");
    printf("\n\t\t\t############                                                   ############");
    printf("\n\t\t\t###########################################################################");
    printf("\n\t\t\t---------------------------------------------------------------------------\n");
    printMessageCenter(message);
    printf("\n\t\t\t----------------------------------------------------------------------------");
}
```

This function displays the welcome message to the user telling him the intro of the program title and it hangs out until the user is ready to continue in the program so, he should press any key to do so.

```c
// This function displays the first welcomes screen of the "SUPER Management System" and asks the user to press any key to access the application.
void welcomeMessage()
{
    headMessage("Zagazig University");
    printf("\n\n\n\n\n");
    printf("\n\t\t\t  **_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**\n");
    printf("\n\t\t\t        =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=");
    printf("\n\t\t\t        =                 WELCOME                   =");
    printf("\n\t\t\t        =                   TO                      =");
    printf("\n\t\t\t        =                 SUPERMARKET               =");
    printf("\n\t\t\t        =               MANAGEMENT                  =");
    printf("\n\t\t\t        =                 SYSTEM                    =");
    printf("\n\t\t\t        =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=");
    printf("\n\t\t\t  **_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**\n");
    printf("\n\n\n\t\t\t Enter any key to continue.....");
    getch();
}
```

This is a helper function used to validate the string input of the user that is used to validate that the user has entered a valid item name or any other data its type is string.

```c
// It validates the user name, item name ..etc. I have permitted this function to take the space in names.
int isNameValid(const char *name)
{
    int validName = 1;
    int len = 0;
    int index = 0;
    len = strlen(name);
    for(index =0; index <len ; ++index)
    {
        if(!(isalpha(name[index])) && (name[index] != '\n') && (name[index] != ' '))

        {
            validName = 0;
            break;
        }
    }
    return validName;
}
```

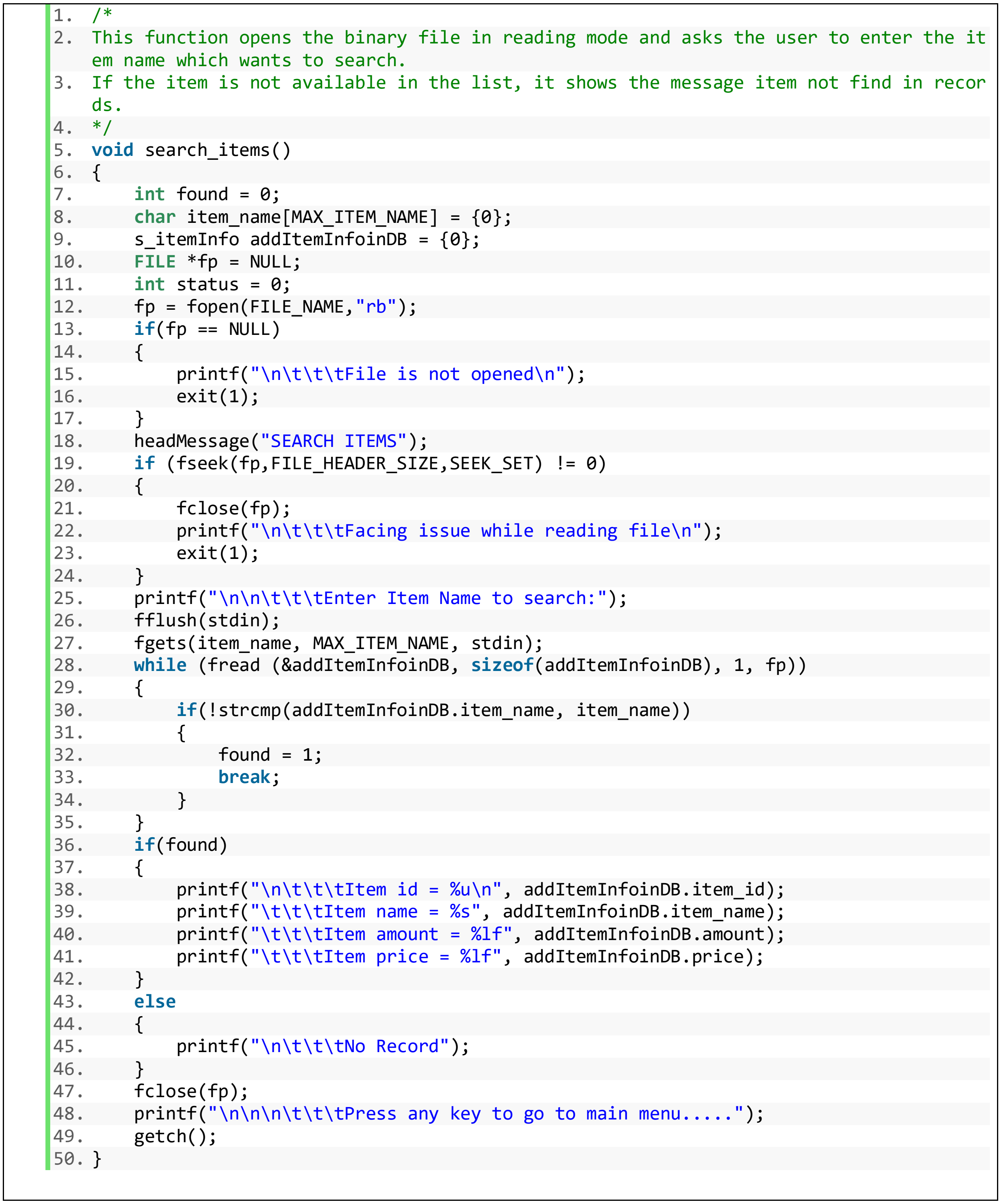
This function helps the add function to do its job by first searching the file for the existence of the item before entering it.

```c
// This function searches the database by item id.
int searchByID(unsigned int itemId){
    int found = 0;
    s_itemInfo addItemInfoinDB = {0};
    FILE *fp = NULL;
    fp = fopen(FILE_NAME,"rb");
    if(fp == NULL)
    {
        printf("\n\t\t\tFile is not opened\n");
        exit(1);
    }
    if (fseek(fp,FILE_HEADER_SIZE,SEEK_SET) != 0)
    {
        fclose(fp);
        printf("\n\t\t\tFacing issue while reading file\n");
        exit(1);
    }

    while (fread (&addItemInfoinDB, sizeof(addItemInfoinDB), 1, fp))
    {
        if(addItemInfoinDB.item_id == itemId)
        {
            found = 1;
            break;
        }
    }

    return found;

}
```

This is the first important function in our program. It used to enter an item by taking the item name, amount, id, and price from the user. But, before doing so it firstly checks the existence of the item in the file. If the item is already found then there is no need to add it again and it displays a message to inform the user. Note that this function uses the last function search by id to perform the process of searching before addition.

```c
// This function opens the binary file in append mode and writes the item and the details.
void add_item()
{
    int days;
    s_itemInfo addItemInfo = {0};
    FILE *fp = NULL;
    int status = 0;
    fp = fopen(FILE_NAME,"ab+");
    if(fp == NULL)
    {
        printf("File is not opened\n");
        exit(1);
    }
    headMessage("ADD NEW ITEMS");
    printf("\n\n\t\t\tENTER YOUR DETAILS BELOW:");
    printf("\n\t\t\t----------------------------------------------------------------------------\n");
    printf("\n\t\t\tItem ID NO  = ");
    fflush(stdin);
    scanf("%u",&addItemInfo.item_id);
    if (searchByID(addItemInfo.item_id)){
            printf("\n\t\t\tItem is already added!!");

    } else {
    do
    {
        printf("\n\t\t\tItem Name  = ");
        fflush(stdin);
        fgets(addItemInfo.item_name, MAX_ITEM_NAME, stdin);
        status = isNameValid(addItemInfo.item_name);
        if (!status)
        {
            printf("\n\t\t\tName contain invalid character. Please enter again.");
        }
    }
    while(!status);
    printf("\n\t\t\tEnter Item Amount  = ");
    fflush(stdin);
    scanf("%lf",&addItemInfo.amount);
    printf("\n\t\t\tEnter Item Price  = ");
    fflush(stdin);
    scanf("%lf",&addItemInfo.price);
    fwrite(&addItemInfo, sizeof(addItemInfo), 1, fp);
    fclose(fp);
    printf("\n\t\t\t------- Item has been added successfully -------");
    }
    printf("\n\t\t\tEnter any key to continue....");
    getch();
}
```

This function is the second function to implement the main functionalities of the program. It searches the file by the item name then displaying it to the user. If the item isn’t found it displays a message to inform the user.



```c
/*
This function opens the binary file in reading mode and asks the user to enter the item name which wants to search.
If the item is not available in the list, it shows the message item not find in records.
*/
void search_items()
{
    int found = 0;
    char item_name[MAX_ITEM_NAME] = {0};
    s_itemInfo addItemInfoinDB = {0};
    FILE *fp = NULL;
    int status = 0;
    fp = fopen(FILE_NAME,"rb");
    if(fp == NULL)
    {
        printf("\n\t\t\tFile is not opened\n");
        exit(1);
    }
    headMessage("SEARCH ITEMS");
    if (fseek(fp,FILE_HEADER_SIZE,SEEK_SET) != 0)
    {
        fclose(fp);
        printf("\n\t\t\tFacing issue while reading file\n");
        exit(1);
    }
    printf("\n\n\t\t\tEnter Item Name to search:");
    fflush(stdin);
    fgets(item_name, MAX_ITEM_NAME, stdin);
    while (fread (&addItemInfoinDB, sizeof(addItemInfoinDB), 1, fp))
    {
        if(!strcmp(addItemInfoinDB.item_name, item_name))
        {
            found = 1;
            break;
        }
    }
    if(found)
    {
        printf("\n\t\t\tItem id = %u\n", addItemInfoinDB.item_id);
        printf("\t\t\tItem name = %s", addItemInfoinDB.item_name);
        printf("\t\t\tItem amount = %lf", addItemInfoinDB.amount);
        printf("\t\t\tItem price = %lf", addItemInfoinDB.price);
    }
    else
    {
        printf("\n\t\t\tNo Record");
    }
    fclose(fp);
    printf("\n\n\n\t\t\tPress any key to go to main menu.....");
    getch();
}
```

This function is used to read all the data from the file and viewing it in a tabular format using printf function and format specifiers.

```c
/*
It opens the file in reading mode and read and display all the stored items details.
If there is no item available in the records, then it displays the message record is empty.
*/
void view_items()
{
    int found = 0;
    char item_name[MAX_ITEM_NAME] = {0};
    s_itemInfo addItemInfoinDB = {0};
    FILE *fp = NULL;
    int status = 0;
    headMessage("VIEW ITEMS DETAILS");
    fp = fopen(FILE_NAME,"rb");
    if(fp == NULL)
    {
        printf("File is not opened\n");
        exit(1);
    }
    if (fseek(fp,FILE_HEADER_SIZE,SEEK_SET) != 0)
    {
        fclose(fp);
        printf("Facing issue while reading file\n");
        exit(1);
    }

    printf("\n\n\nItem ID\t\t\tItem Amount\t\t\tItem Price\t\t\tItem Name\n");

    while (fread (&addItemInfoinDB, sizeof(addItemInfoinDB), 1, fp))
    {
        printf("%u\t\t\t%lf\t\t\t%lf\t\t\t%s\n", addItemInfoinDB.item_id, addItemInfoinDB.amount, addItemInfoinDB.price, addItemInfoinDB.item_name);
        found = 1;
    }
    fclose(fp);
    if(!found)
    {
        printf("\n\t\t\tNo Record");
    }
    printf("\n\n\t\t\tPress any key to go to main menu.....");
    fflush(stdin);
    getch();
}
```

The main role for this item is to access the file and read data based on the item id that the user will enter. Using this information to print finally information like a bill in the supermarket by multiplying the price of the item by the amount that the shopper has bought.

```c
// This function access the items by id and get its price to complete the payment.
void payItems(){
    int found = 0;
    char item_name[MAX_ITEM_NAME] = {0};
    s_itemInfo addItemInfoinDB = {0};
    FILE *fp = NULL;
    int status = 0;
    fp = fopen(FILE_NAME,"rb");
    if(fp == NULL)
    {
        printf("\n\t\t\tFile is not opened\n");
        exit(1);
    }
    headMessage("PAY ITEMS");
    if (fseek(fp,FILE_HEADER_SIZE,SEEK_SET) != 0)
    {
        fclose(fp);
        printf("\n\t\t\tFacing issue while reading file\n");
        exit(1);
    }
    unsigned int itemid = 0;
    char another = 'Y';
    while (another == 'y' || another == 'Y'){
    printf("\n\n\t\t\tEnter Item ID to pay:");
    fflush(stdin);
    scanf("%u", &itemid);
    while (fread (&addItemInfoinDB, sizeof(addItemInfoinDB), 1, fp))
    {
        if(addItemInfoinDB.item_id == itemid)
        {
            found = 1;
            break;
        }
    }
    if(found)
    {
            printf("\n\t\t\tPlease Enter The needed amount: ");
            double am = 1;
            scanf("%lf", &am);
            printf("\n\t\t\tItem name: %s\n", addItemInfoinDB.item_name);
            printf("\n\t\t\tItem price: %lf\n", (addItemInfoinDB.price) * am);
    }
    else
    {
        printf("\n\t\t\tNo Record\n");
    }

        printf("\n\t\t\tPay another Item?(Y/N)");
        fflush(stdin);
        another=getchar();
    }

    fclose(fp);
    printf("\n\n\n\t\t\tPress any key to go to main menu.....");
    getch();

}
```

This function is used to delete the item based in its name, the user will enter the name of the item and then a while loop will seeks the file until the desired item is found by moving the needed data to another file and deleting the existing file.

```c
/*
This function asks the item name from the user for the item want to delete.
In this function, I am creating a temporary binary file and copy all the data from the existing file except the item whose item name entered by the user.
In the last renamed the temporary bin file with an existing binary file.
*/
void delete_items()
{
    int found = 0;
    sFileHeader fileHeaderInfo = {0};
    char item_name[MAX_ITEM_NAME] = {0};
    s_itemInfo addItemInfoinDB = {0};
    FILE *fp = NULL;
    FILE *tmpFp = NULL;
    int status = 0;
    headMessage("Delete Items Details");
    fp = fopen(FILE_NAME,"rb");
    if(fp == NULL)
    {
        printf("File is not opened\n");
        exit(1);
    }
    tmpFp = fopen("tmp.bin","wb");
    if(tmpFp == NULL)
    {
        fclose(fp);
        printf("File is not opened\n");
        exit(1);
    }
    fread (&fileHeaderInfo,FILE_HEADER_SIZE, 1, fp);
    fwrite(&fileHeaderInfo,FILE_HEADER_SIZE, 1, tmpFp);
    printf("\n\t\t\tEnter Item Name for delete:");
    fflush(stdin);
    fgets(item_name, MAX_ITEM_NAME, stdin);
    while (fread (&addItemInfoinDB, sizeof(addItemInfoinDB), 1, fp))
    {
        if(strcmp(addItemInfoinDB.item_name, item_name))
        {
            fwrite(&addItemInfoinDB, sizeof(addItemInfoinDB), 1, tmpFp);
        }
        else
        {
            found = 1;
        }
    }
    (found)? printf("\n\t\t\tRecord deleted successfully......"):printf("\n\t\t\tRecord not found");
    fclose(fp);
    fclose(tmpFp);
    remove(FILE_NAME);
    rename("tmp.bin",FILE_NAME);
}
```

This function update the info for the item. The user should enter the item name to access its data and change it.

```c
/*
This function update the info for the item. the user should enter the item name to access its data and change it.
*/
void update_itemInfo(){
    int status = 0;
    int found = 0;
    sFileHeader fileHeaderInfo = {0};
    char item_name[MAX_ITEM_NAME] = {0};
    s_itemInfo addItemInfo = {0};
    FILE *fp = NULL;
    headMessage("Update Items Details");
    fp = fopen(FILE_NAME,"rb+");
    if(fp == NULL)
    {
        printf("File is not opened\n");
        exit(1);
    }
    fread (&fileHeaderInfo,FILE_HEADER_SIZE, 1, fp);
    printf("\n\t\t\tEnter Item Name for update: ");
    fflush(stdin);
    fgets(item_name, MAX_ITEM_NAME, stdin);
    while (fread (&addItemInfo, sizeof(addItemInfo), 1, fp))
    {
        if(!strcmp(addItemInfo.item_name, item_name))
        {
            fflush(stdin);
            printf("\n\t\t\tThe item is available\n");
            found = 1;
            printf("\n\t\t\tEnter new item id: ");
            fflush(stdin);
            scanf("%u",&addItemInfo.item_id);

            do{
            printf("\n\t\t\tEnter new Item name: ");
            fflush(stdin);
            fgets(addItemInfo.item_name, MAX_ITEM_NAME, stdin);
            status = isNameValid(addItemInfo.item_name);
            if (!status)
            {
                printf("\n\t\t\tName contain invalid character. Please enter again.");
            }
            }while(!status);

            printf("\n\t\t\tEnter new Item Amount  = ");
            fflush(stdin);
            scanf("%lf",&addItemInfo.amount);
            printf("\n\t\t\tEnter new Item Price  = ");
            fflush(stdin);
            scanf("%lf",&addItemInfo.price);
            fseek(fp,ftell(fp)-sizeof(addItemInfo),0);
            fwrite(&addItemInfo, sizeof(addItemInfo), 1, fp);
            printf("\n\t\t\t------- Item has been updated successfully -------\n\n");
            break;
        }
    }
    fclose(fp);
    if (!found){
        printf("\n\t\t\tThe record is not available!!");
    }
    printf("\n\t\t\tEnter any key to continue....");
    getch();

}
```

This function opens the file in rb+ mode (reading and writing). It asks the user for the new username and password. After taking the password and username it closes the application. Now user can use the application with a new password and username.

```c
/*
This function opens the file in rb+ mode (reading and writing).
It asks the user for the new username and password.
After taking the password and username it closes the application.
Now user can use the application with a new password and username.
*/
void update_loginInfo(void)
{
    sFileHeader fileHeaderInfo = {0};
    FILE *fp = NULL;
    unsigned char userName[MAX_SIZE_USER_NAME] = {0};
    unsigned char password[MAX_SIZE_PASSWORD] = {0};
    headMessage("Update Credential");
    fp = fopen(FILE_NAME,"rb+");
    if(fp == NULL)
    {
        printf("File is not opened\n");
        exit(1);
    }
    fread (&fileHeaderInfo,FILE_HEADER_SIZE, 1, fp);
    if (fseek(fp,0,SEEK_SET) != 0)
    {
        fclose(fp);
        printf("\n\t\t\tFacing issue while updating password\n");
        exit(1);
    }
    printf("\n\n\t\t\tNew Username:");
    fflush(stdin);
    fgets(userName,MAX_SIZE_USER_NAME,stdin);
    printf("\n\n\t\t\tNew Password:");
    fflush(stdin);
    fgets(password,MAX_SIZE_PASSWORD,stdin);
    strncpy(fileHeaderInfo.username,userName,sizeof(userName));
    strncpy(fileHeaderInfo.password,password,sizeof(password));
    fwrite(&fileHeaderInfo,FILE_HEADER_SIZE, 1, fp);
    fclose(fp);
    printf("\n\t\t\tYour Password has been changed successfully");
    printf("\n\t\t\tLogin Again:");
    fflush(stdin);
    getch();
    exit(1);
}
```

This function displays the application menu and asks the user to select the option. If the user selects 0, then the application will close.

```c
// This function displays the application menu and asks the user to select the option. If the user selects 0, then the application will close.

void menu()
{
    int choice = 0;
    do
    {
        headMessage("MAIN MENU");
        printf("\n\n\n\t\t\t1.Add Items");
        printf("\n\t\t\t2.Search Items");
        printf("\n\t\t\t3.View Items");
        printf("\n\t\t\t4.Pay for shopper");
        printf("\n\t\t\t5.Delete Items");
        printf("\n\t\t\t6.Update Items Info");
        printf("\n\t\t\t7.Update Password");
        printf("\n\t\t\t0.Exit");
        printf("\n\n\n\t\t\tEnter choice => ");
        scanf("%d",&choice);
        switch(choice)
        {
            case 1:
                add_item();
                break;
            case 2:
                search_items();
                break;

            case 3:
                view_items();
                break;
            case 4:
                payItems();
                break;
            case 5:
                delete_items();
                break;
            case 6:
                update_itemInfo();
                break;
            case 7:
                update_loginInfo();
                break;
            case 0:
                printf("\n\n\n\t\t\tThank you!!!\n\n\n\n\n");
                exit(1);
                break;
            default:
                printf("\n\n\n\t\t\tINVALID INPUT!!! Try again...");
        }                                            //Switch Ended
    }
    while(choice!=0);                                        //Loop Ended
}
```

This function is used to check the authority of the program user, it handles the login process to the system using a username and a password. If the username and password are wrong for 3 time, then the program will exit.

```
//login password
void login()
{
    unsigned char userName[MAX_SIZE_USER_NAME] = {0};
    unsigned char password[MAX_SIZE_PASSWORD] = {0};
    int L=0;
    sFileHeader fileHeaderInfo = {0};
    FILE *fp = NULL;
    headMessage("Login");
    fp = fopen(FILE_NAME,"rb");
    if(fp == NULL)
    {
        printf("File is not opened\n");
        exit(1);
    }
    fread (&fileHeaderInfo,FILE_HEADER_SIZE, 1, fp);
    fclose(fp);
    do
    {
        printf("\n\n\n\t\t\t\tUsername:");
        fgets(userName,MAX_SIZE_USER_NAME,stdin);
        printf("\n\t\t\t\tPassword:");
        fgets(password,MAX_SIZE_PASSWORD,stdin);
        if((!strcmp(userName,fileHeaderInfo.username)) && (!strcmp(password,fileHeaderInfo.password)))
        {
            menu();
        }
        else
        {
            printf("\t\t\t\tLogin Failed Enter Again Username & Password\n\n");
            L++;
        }
    }
    while(L<=3);
    if(L>3)
    {
        headMessage("Login Failed");
        printf("\t\t\t\tSorry,Unknown User.");
        getch();
        system("cls");
    }
}
```

This function verifies that a file has been created or not. If the file exists, the function return 1 otherwise returns 0.

```c
// This function verifies that a file has been created or not. If the file exists, the function return 1 otherwise returns 0.
int isFileExists(const char *path)
{
    // Try to open file
    FILE *fp = fopen(path, "rb");
    int status = 0;
    // If file does not exists
    if (fp != NULL)
    {
        status = 1;
        // File exists hence close file
        fclose(fp);
    }
    return status;
}
```

This function creates the file if not exist and copies the default password ("admin") in file header structure.

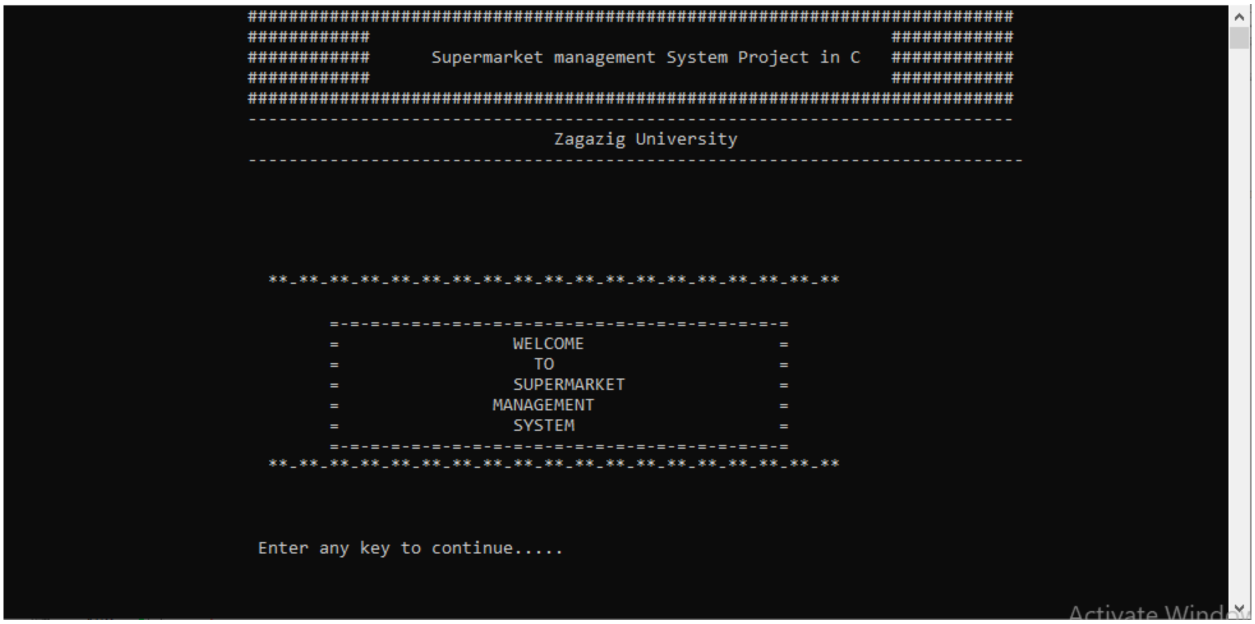
```c
// This function creates the file if not exist and copies the default password ("admin") in file header structure.
void init()
{
    FILE *fp = NULL;
    int status = 0;
    const char defaultUsername[] ="admin\n";
    const char defaultPassword[] ="admin\n";
    sFileHeader fileHeaderInfo = {0};
    status = isFileExists(FILE_NAME);
    if(!status)
    {
        //create the binary file
        fp = fopen(FILE_NAME,"wb");
        if(fp != NULL)
        {
            //Copy default password
            strncpy(fileHeaderInfo.password,defaultPassword,sizeof(defaultPassword));
            strncpy(fileHeaderInfo.username,defaultUsername,sizeof(defaultUsername));
            fwrite(&fileHeaderInfo,FILE_HEADER_SIZE, 1, fp);
            fclose(fp);
        }
    }
}
```

The main function where the program starts is execution. It invokes the appropriate functions for runtime of the program.

```
int main()
{
    init();
    welcomeMessage();
    login();
    return 0;
}
```

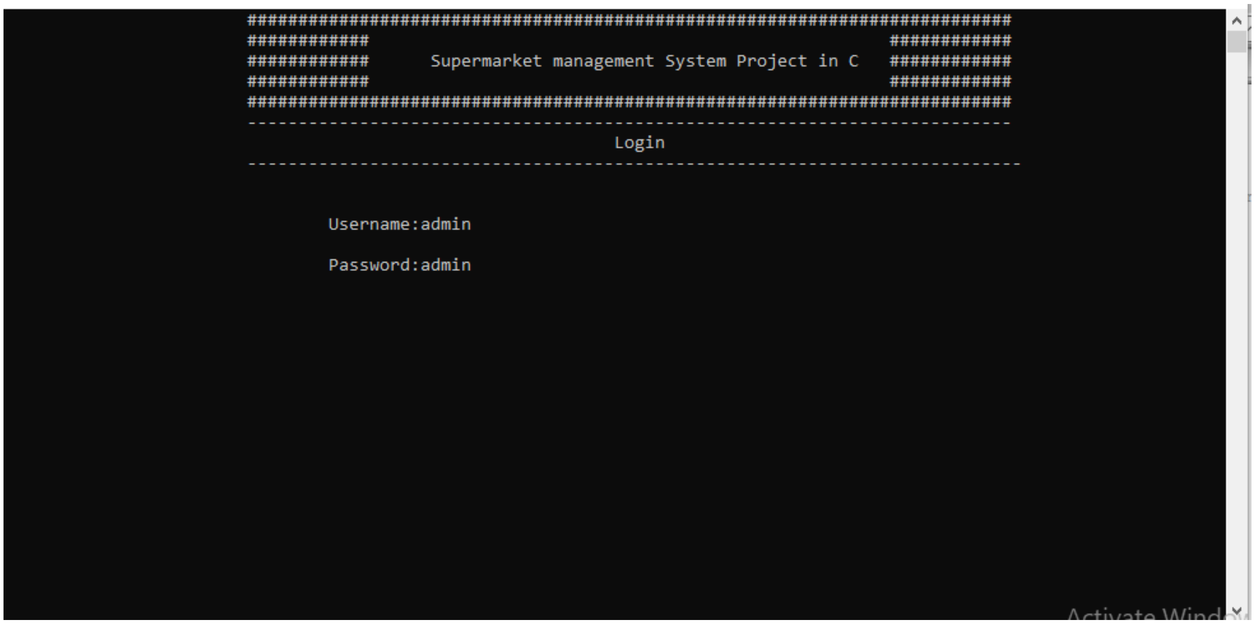
Screenshots

This figure shows the intro screen for the program.



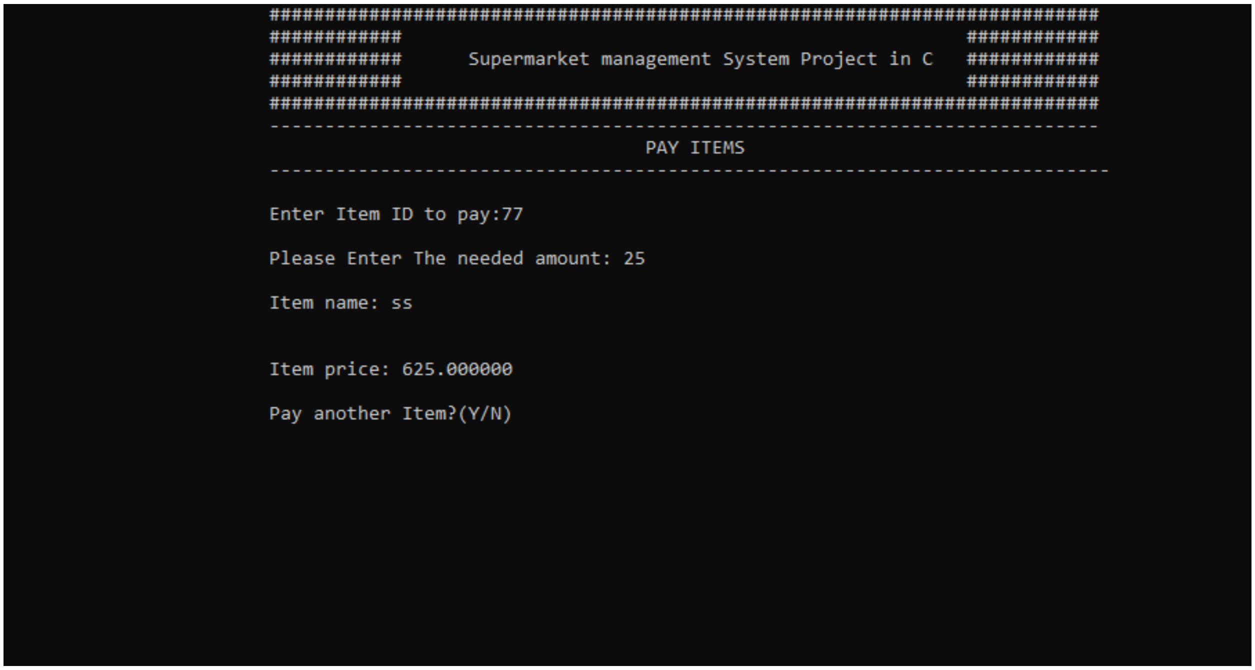
```
###########################################################################
############                                                   ############
############      Supermarket management System Project in C   ############
############                                                   ############
###########################################################################
---------------------------------------------------------------------------
                              Zagazig University
----------------------------------------------------------------------------




  **_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**

        =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
        =                 WELCOME                   =
        =                   TO                      =
        =                 SUPERMARKET               =
        =               MANAGEMENT                  =
        =                 SYSTEM                    =
        =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
  **_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**_**



 Enter any key to continue.....
```

This figure shows the login screen for the program. The default username and password are “admin” and “admin” respectively.



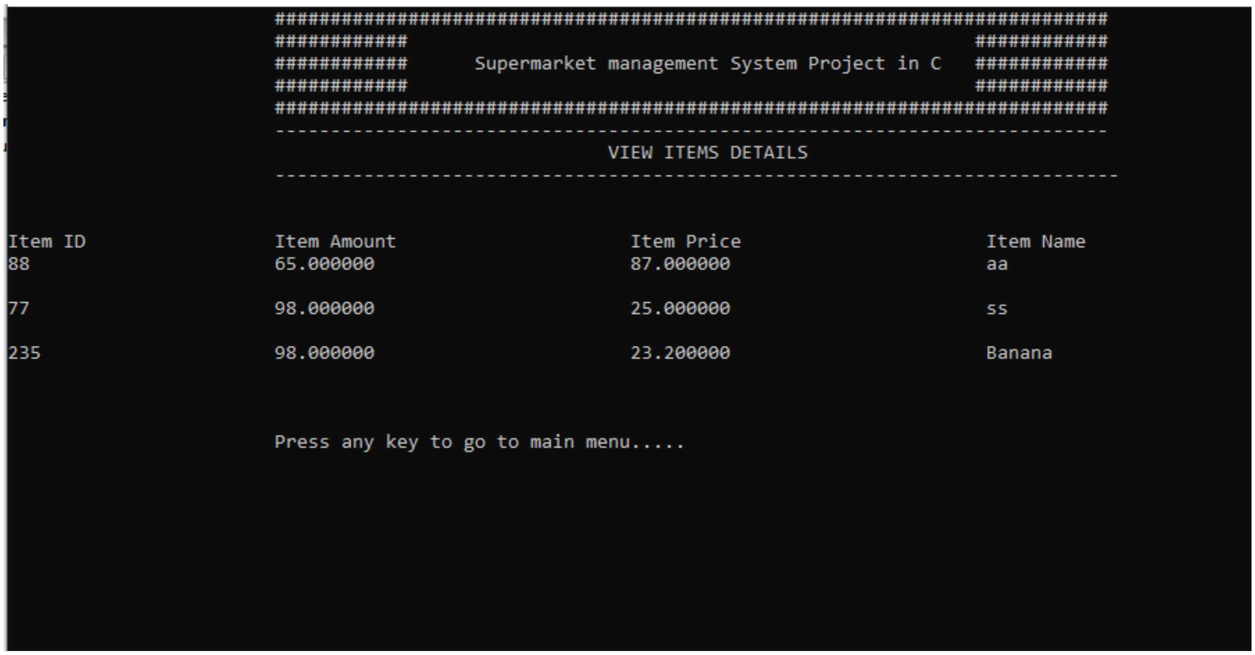
```
############################################################################
############                                                    ############
############      Supermarket management System Project in C    ############
############                                                    ############
############################################################################
----------------------------------------------------------------------------
                                    Login
-----------------------------------------------------------------------------


        Username:admin

        Password:admin
```

When the user chooses to pay for an item.



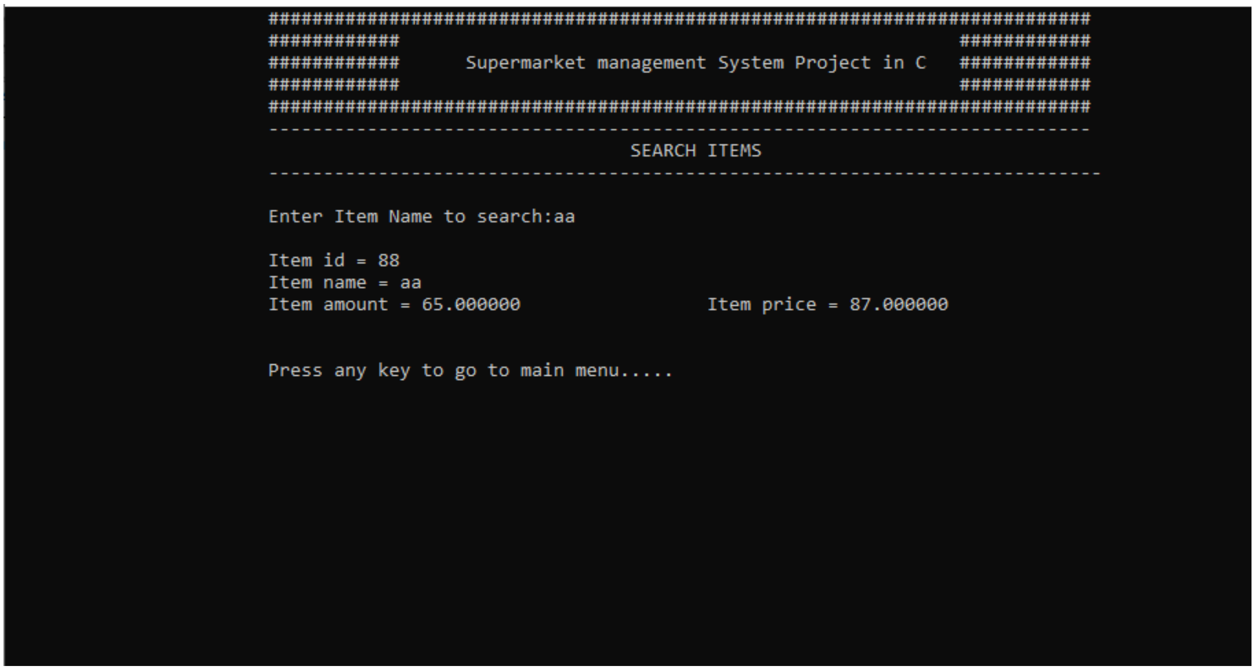
```
############################################################################
############                                                    ############
############      Supermarket management System Project in C    ############
############                                                    ############
############################################################################
----------------------------------------------------------------------------
                                  PAY ITEMS
-----------------------------------------------------------------------------

Enter Item ID to pay:77

Please Enter The needed amount: 25

Item name: ss


Item price: 625.000000

Pay another Item?(Y/N)
```

The result of viewing all items in tabular format.



```
                        ###########################################################################
                        ############                                                  ############
                        ############      Supermarket management System Project in C   ############
                        ############                                                  ############
                        ###########################################################################
                        ---------------------------------------------------------------------------
                                                      VIEW ITEMS DETAILS
                        ----------------------------------------------------------------------------


Item ID                 Item Amount                     Item Price                      Item Name
88                      65.000000                       87.000000                       aa

77                      98.000000                       25.000000                       ss

235                     98.000000                       23.200000                       Banana



                        Press any key to go to main menu.....
```

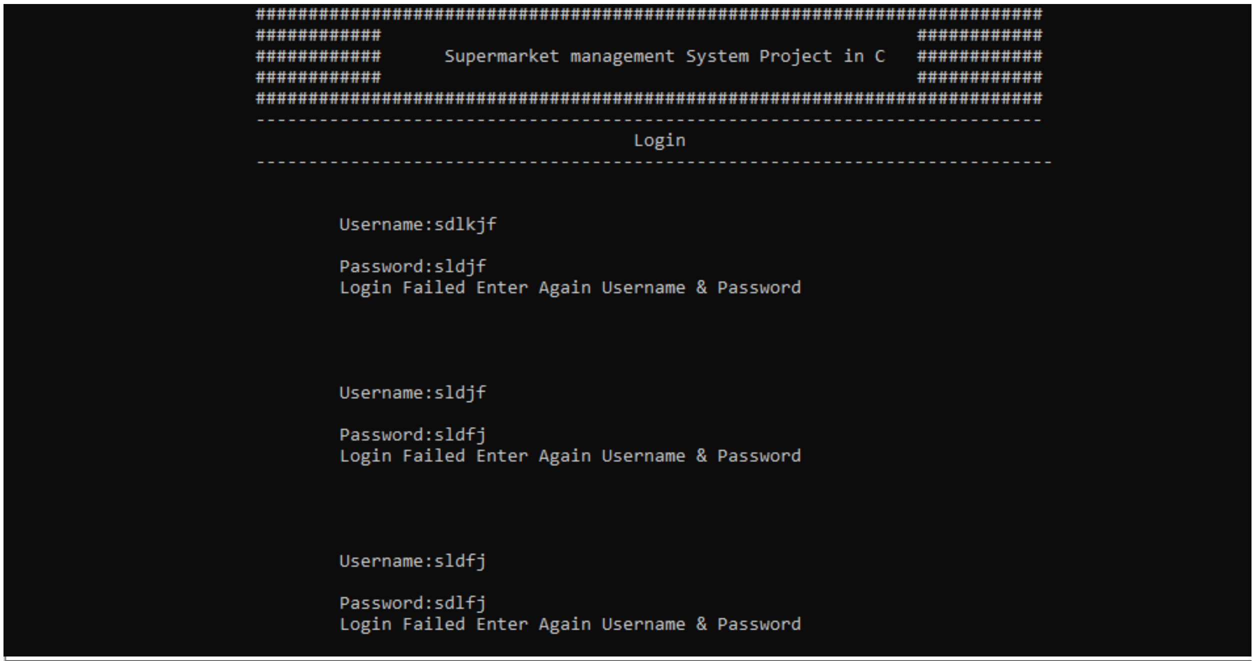
When the user searches for an item using its name.



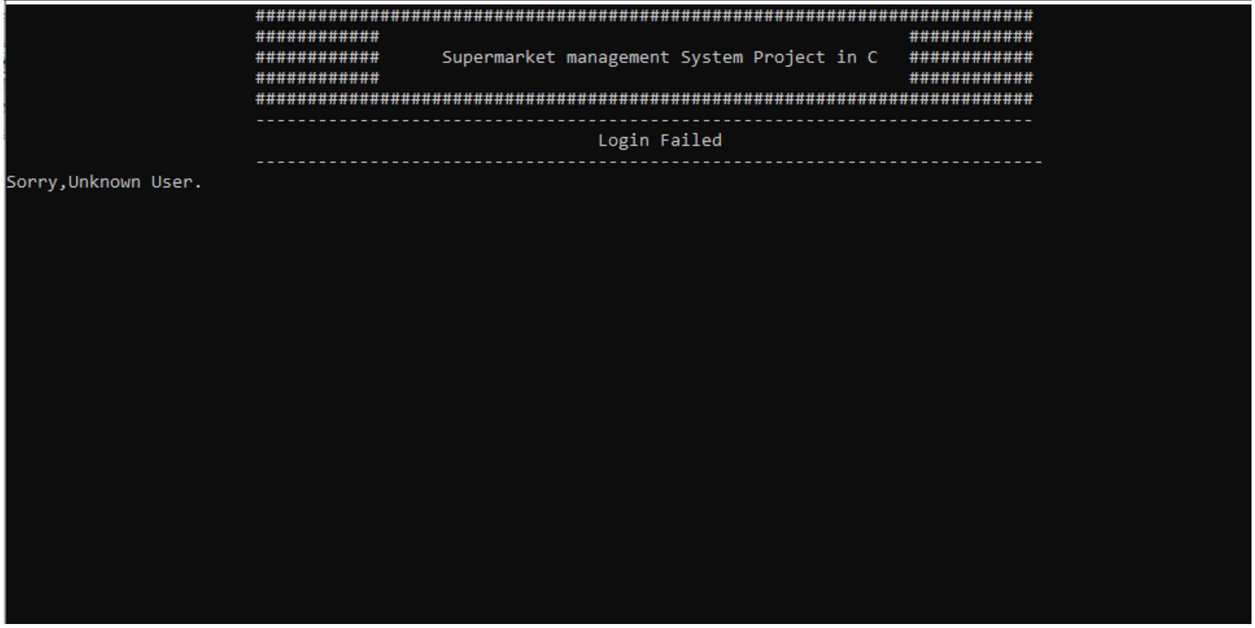
```
                        ###########################################################################
                        ############                                                  ############
                        ############      Supermarket management System Project in C   ############
                        ############                                                  ############
                        ###########################################################################
                        ---------------------------------------------------------------------------
                                                        SEARCH ITEMS
                        ----------------------------------------------------------------------------

                        Enter Item Name to search:aa

                        Item id = 88
                        Item name = aa
                        Item amount = 65.000000                 Item price = 87.000000


                        Press any key to go to main menu.....
```

Important Discussions

When the user enters the username and password incorrectly for 3 times, the program displays an error and exits.

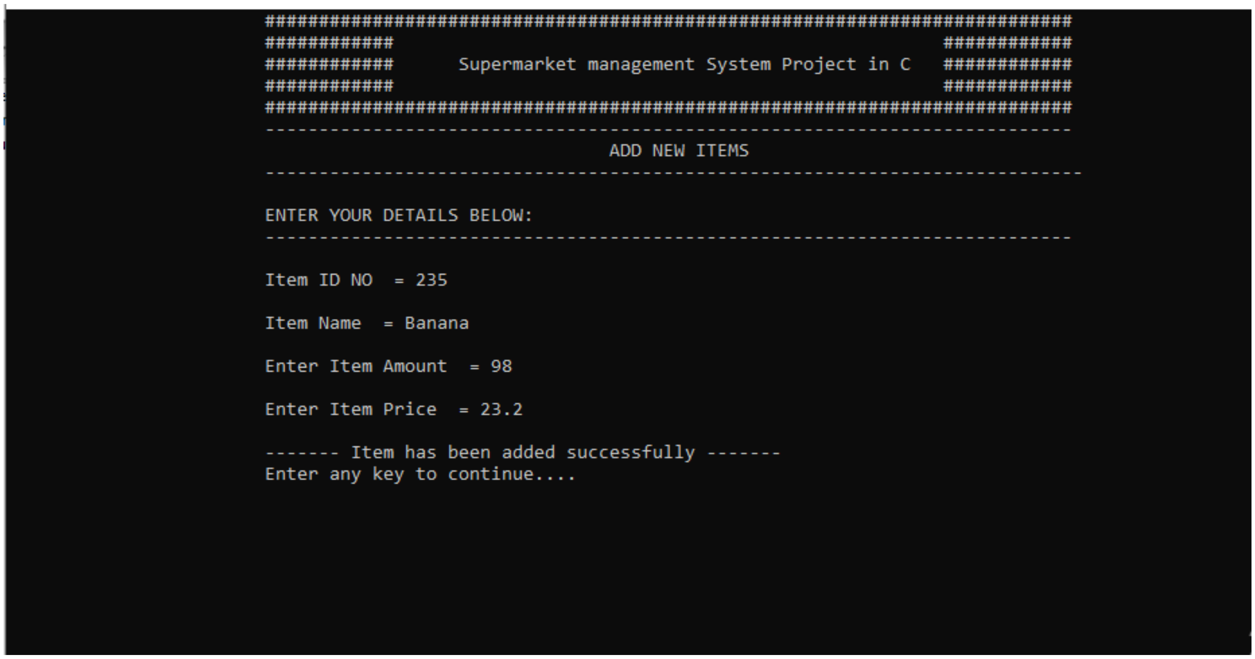


```
                        ############################################################################
                        ############                                                    ############
                        ############      Supermarket management System Project in C    ############
                        ############                                                    ############
                        ############################################################################
                        ----------------------------------------------------------------------------
                                                    Login
                        -----------------------------------------------------------------------------


                                Username:sdlkjf

                                Password:sldjf
                                Login Failed Enter Again Username & Password



                                Username:sldjf

                                Password:sldfj
                                Login Failed Enter Again Username & Password



                                Username:sldfj

                                Password:sdlfj
                                Login Failed Enter Again Username & Password
```

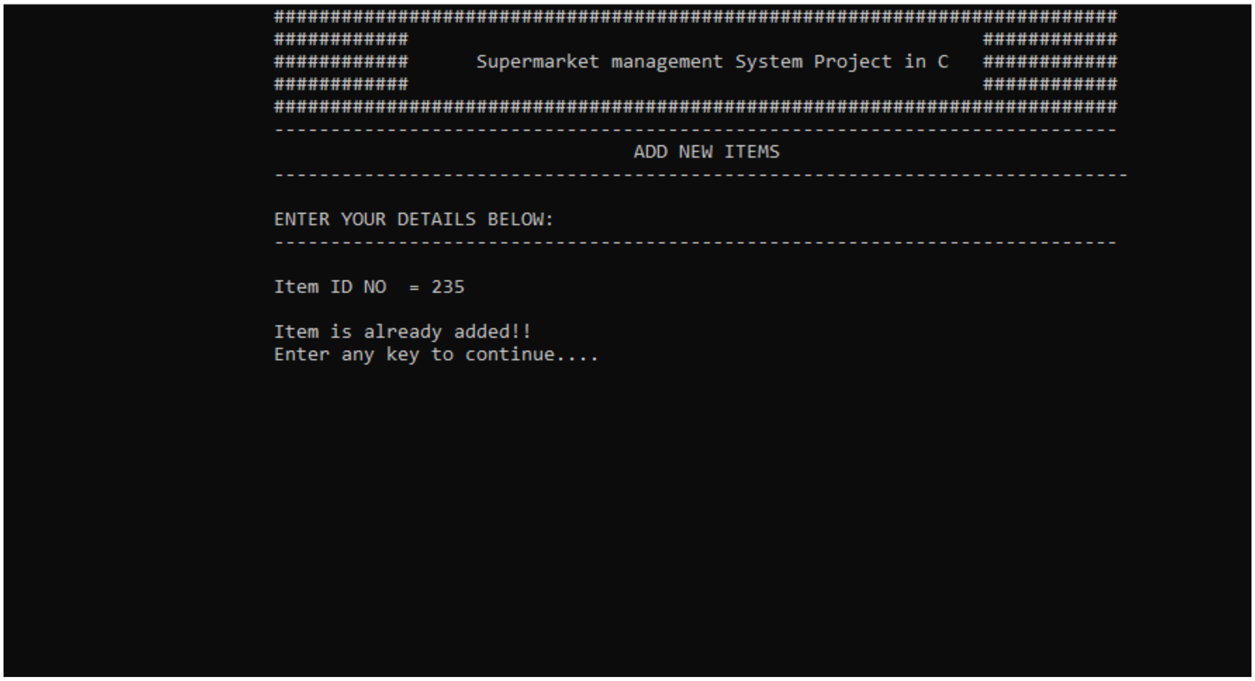


```
                        ###########################################################################
                        ############                                                   ############
                        ############      Supermarket management System Project in C   ############
                        ############                                                   ############
                        ###########################################################################
                        ---------------------------------------------------------------------------
                                                  Login Failed
                        ----------------------------------------------------------------------------
Sorry,Unknown User.
```

When the user chooses to add items it firstly searches its existence if it is found the user couldn't add it again.



```
###########################################################################
############                                                   ############
############      Supermarket management System Project in C   ############
############                                                   ############
###########################################################################
---------------------------------------------------------------------------
                                ADD NEW ITEMS
----------------------------------------------------------------------------

ENTER YOUR DETAILS BELOW:
---------------------------------------------------------------------------

Item ID NO  = 235

Item Name  = Banana

Enter Item Amount  = 98

Enter Item Price  = 23.2

------- Item has been added successfully -------
Enter any key to continue....
```



```
###########################################################################
############                                                   ############
############      Supermarket management System Project in C   ############
############                                                   ############
###########################################################################
---------------------------------------------------------------------------
                                ADD NEW ITEMS
----------------------------------------------------------------------------

ENTER YOUR DETAILS BELOW:
---------------------------------------------------------------------------

Item ID NO  = 235

Item is already added!!
Enter any key to continue....
```

Conclusion

The program overall is important to learn about structs and pointers. Also, it uses file handling intensively. The main functions for CRUD uses file handling instead of databases in order to save data.

One recommendation is to use a database instead of files because they are more robust.

References

[1] Kernighan, Brian W., and Dennis M. Ritchie. The C Programming Language. Prentice Hall, 2016.

[2] “C Language.” Cppreference.com, en.cppreference.com/w/c/language.

[3] “C Tutorial.” Tutorialspoint, www.tutorialspoint.com/cprogramming/index.htm.