**INVENTORY MANAGEMENT SYSTEM**

## Submitted by

**Name of the Student:** Agnik Maity

**Enrollment Number:** 12022002003106

**Section:** G

**Class Roll Number:** 08

**Stream:** ECE

**Subject:** Programming for Problem Solving using C

**Subject Code:** ESC 103 (Pr.)

**Department:** Basic Science and Humanities

**Academic Year: 2022-26**



**DEPARTMENT OF BASIC SCIENCE AND HUMANITITES INSTITUTE OF ENGINEERING AND MANAGEMENT, KOLKATA**

**PROGRAM NAME:**

Create a program that manages inventory, including tracking product sales, managing stock, and generating reports.

**PROGRAM CODE:**

#include<stdio.h>

#include<string.h>//strlen

#include <stdlib.h>

#define true 0 //true boolean

#define false 1 // for false the boolean

struct product

{

char id[10]; // product code/no.

char name[20]; // product name

int quantity; // remaining quantity of product. Subtract from the original quantity the quantity purchased

int numSold; // initially zero, when no purchase yet.

float price; // price of one piece of product

int discount; // discount for this product

float sales; // accumulated sales, total sales for this product

};

struct product prod[30];//the maximum array elements.

int count = 0; // this will be incremented if there is a new product and this is the

FILE \*f; //for the file pointer

int writefile()//write file function

{

int i;

f = fopen("inventory.txt", "w"); // ayaw i append; change from f = fopen("inventory.txt", "a");

if (f == NULL)

return -1;

fprintf(f, "%d\n", count);

for (i = 0; i < count; ++i) // writing all the details from all the function to the text file.

{

// Changed

fputs(prod[i].id, f);

fprintf(f, "\n");

fputs(prod[i].name, f);

fprintf(f, "\n");

fprintf(f, "%d\n", prod[i].quantity);

fprintf(f, "%d\n", prod[i].numSold);

fprintf(f, "%f\n", prod[i].price);

fprintf(f, "%d\n", prod[i].discount);

fprintf(f, "%f\n", prod[i].sales);

}

fclose(f);

return 0;

}

int readFile() // read file function

{

int n = 0;

int i;

f = fopen("inventory.txt", "r");

if (f == NULL)

return -1;

fscanf(f, "%d\n", &n);

for (i = 0; i < n; ++i)

{

fgets(prod[i].id, 10, f);

prod[i].id[strlen(prod[i].id) - 1] = 0; // remove new lines

fgets(prod[i].name, 20, f);

prod[i].name[strlen(prod[i].name)-1] = 0; // remove new lines

fscanf(f, "%d", &prod[i].quantity);

fscanf(f, "%d", &prod[i].numSold);

fscanf(f, "%f", &prod[i].price);

fscanf(f, "%d", &prod[i].discount);

fscanf(f, "%f\n", &prod[i].sales);

}

fclose(f);

return n;

}

void disZeroQuant(){ // for the switch number 7= calling all the product id with zero quantity.

int i;

count= readFile(); // call the read function

printf("\nProducts with zero Quantity: ");

for (i=0; i<count; i++){

if(prod[i].quantity==0){// printing the highest product.

printf("\nName of the product: %s \nProduct Id: %s \nQuantity left: %d \nNumber of product sold: %d \nPrice of the product: %.2f \nDiscount of the product: %d %% \nTotal Sales: %.2lf\n",prod[i].name,prod[i].id,prod[i].quantity,prod[i].numSold,prod[i].price,prod[i].discount,prod[i].sales);

}

}

writefile();

}

void dispHsale(){ // to display the highest sale function

int high,i;

high=prod[0].numSold; // getting the first element of the array that has been sold

for(i=0;i<count;i++) // loop for the num item sold.

{

if(prod[i].numSold>high) //if the element containts the highest sold product.

high=prod[i].numSold;//it will pass on the high variable.

}

printf("\nThe Highest Product Sale is: \n");

for(i=0;i<count;i++) // loop to search the highest sold product.

{

if(prod[i].numSold==high)// printing the highest product.

printf("Name of the product: %s \nProduct Id: %s \nQuantity left: %d \nNumber of product sold: %d \nPrice of the product: %.2f \nDiscount of the product: %d %% \nTotal Sales: %.2lf\n",prod[i].name,prod[i].id,prod[i].quantity,prod[i].numSold,prod[i].price,prod[i].discount,prod[i].sales);

}

}

void purchaseprod(){// function for purchasing a product

int quant,i;

char id[10];

int z=false;

count=readFile();

printf("Sell an Item ");

printf("\nProduct ID: ");

fflush(stdin);

gets(id);

for (i=0; i<count; i++){

if (strcmp(id,prod[i].id)==0) // if the id that the user want to find and the data id that has been saved at file is matched.

{

z=true;

printf("\nItem found! Containing: \n");//...then display the match

printf("\nProduct name: %s",prod[i].name);

printf("\nPrice: %.2lfphp\n\n",prod[i].price);

printf("Enter the quantity you want to buy : ");

fflush(stdin);

scanf("%d",&quant);

if (quant>prod[i].quantity){ // if the quantity is lessthan the users quant

puts("\nInsufficient Quantity\nPlease Restock.\n ");

break; // break and back to the choices.

}

float tempSales = prod[i].sales; // will be executed if the quantity is greater than the users selected quantity.

prod[i].numSold += quant;

prod[i].quantity -= quant;

prod[i].sales = quant\*(prod[i].price\*(prod[i].discount/100.0));

prod[i].sales += tempSales;

}

}

if(z==false){ //if the product id is not available.

printf("Cant find the product id: %s.",id);

}

writefile();

}

void deleteprod(){ //function for the delete product.

count=readFile();

char id[10];

int i,j;

int z=false;

printf("Enter the id that you want to be delete : "); //user's input for deleting.

fflush(stdin);

gets(id);

for(i=0;i<count;i++){ //loop to finding the user's input

z=true;

if(strcmp(prod[i].id,id)==0){ // if the user's input matched the data

for( j=i; j<(count-1); j++) // it will erase the selected data.

{

prod[j]=prod[j+1];

}

count--;

}

}

if(z==false){ // will be executed if the product id is not available.

printf("Cant find product id: %s .",id);

}

writefile();

}

void addProd(){ // function to add products to the file

printf("ENTER NEW PRODUCTS\n");

readFile(); //reading the files .

if (count>0) {

count=readFile();

IDChecker(0,count); // to check if the id is already used.

}

else{

printf("\nProduct ID Number: ");

fflush(stdin);

gets(prod[count].id);

}

printf("Product Name: ");gets(prod[count].name);

printf("Quantity of the product: ");scanf("%d",&prod[count].quantity);

printf("Price of the product: ");scanf("%f",&prod[count].price);

printf("Item Discount: ");scanf("%d",&prod[count].discount);

++count; // increment count for the product positions and how many are they in the array.

writefile(); // putting/saving this to the file.

}

int IDChecker(int i, int j) //checking the input id

{

count=readFile();

printf("Product ID: ");

fflush(stdin);

gets(prod[count].id);

if (strcmp(prod[i].id,prod[j].id)==0){

printf("ID number is already taken!");

return IDChecker(i++,j--);

}

}

void editProd(){ //Editing the product function

char id[10];

int test;

int i;

int choice;

printf("EDIT A PRODUCT!");

printf("\nEnter the id of the product that you want to edit: "); // users input for what data will be change

fflush(stdin);

gets(id);

test=checkID(id);

if (test == 0)

{

printf("The id num %s is not found.", id); // if the data is empty

}

else

{

readFile();

{

for(i=0;i<count;i++){

if(strcmp(id,prod[i].id)!=0) // if the data is not empty

writefile();

else

{

printf("\n1. Update product ID Number?");

printf("\n2. Update Name of the product? ");

printf("\n3. Update Quantitiy of the product?");

printf("\n4. Update Price of the product?");

printf("\n5. Update Discount of the product?");

printf("\nEnter your choice:");

fflush(stdin);

scanf("%d", &choice);

switch (choice)

{

case 1:

printf("Enter new ID: ");

fflush(stdin);

gets(prod[i].id);

break;

case 2:

printf("Enter new Name: ");

fflush(stdin);

gets(prod[i].name);

break;

case 3:

printf("Enter Quantity: ");

scanf("%d",&prod[i].quantity);

break;

case 4:

printf("Enter the new price: ");

scanf("%f",&prod[i].price);

break;

case 5:

printf("Enter the new discount of the product: ");

scanf("%d",&prod[i].discount);

default:

printf("Invalid Selection");

break;

}

writefile();

}

}

}

fclose(f);

f = fopen("Inventory.txt", "r");

readFile();

{

writefile();

}

fclose(f);

printf("RECORD UPDATED");

}

}

int checkID(char id[]){ // checking the id if available

int i;

count=readFile();

readFile();

for(i=0;i<count;i++){

if(strcmp(id,prod[i].id)!=0){ //if the id and data id doesnt match.

fclose(f);

}

return 1; // returning an error.

}

fclose(f);

return 0; // return 0 if no error.

}

void displayprod(){

int i;

count = readFile(); // the output is how many products inside the file.

if (count < 0)

puts("cannot open file");

printf(" \t\t\t\t \*\*\*\*\* INVENTORY \*\*\*\*\*\n");

printf("------------------------------------------------------------------------------------\n");

printf("S.N.| NAME | PROD ID | QUANTITY | PROD SOLD | PRICE | DISCOUNT | SALES |\n");

printf("------------------------------------------------------------------------------------\n");

for (i=0;i<count;i++){ // getting the details on each product updates.

printf("%d %-10s %-8s %-5d %-3d %-6.2f %-5d%% P%.2lf\n",i+1,prod[i].name,prod[i].id,prod[i].quantity,prod[i].numSold,prod[i].price,prod[i].discount,prod[i].sales);

}

}

int main (){

int choice;

count = readFile();

if(count < 0) // there is no file located.

printf("Cannot locate file\n");

do {

printf("\n");

printf("\t\t\t ================================\n");

printf("\t\t\t PRODUCT INVENTORY PROGRAM\n");

printf("\t\t\t ================================");

printf("\n\nPress:");

printf("\n 1.) Input new product record.");

printf("\n 2.) Edit a Product.");

printf("\n 3.) Delete a Product");

printf("\n 4.) Display all existing product.");

printf("\n 5.) Make a purchase.");

printf("\n 6.) Display the product record with highest sale.");

printf("\n 7.) Display all product with zero quantity");

printf("\n 8.) Exit the program.");

printf("\nChoice--> ");

scanf("%d", &choice);

switch(choice){

case 1 : //add product

addProd();

break;

case 2://edit data product

editProd();

break;

case 3://delete a product

deleteprod();

break;

case 4: //display the products

displayprod();

break;

case 5://make a purchased.

purchaseprod();

break;

case 6:

dispHsale(); // to display highest sale.

break;

case 7:

disZeroQuant(); // display lowest sale.

break;

case 8:

exit(1);

break;

default :

printf("Your choice is wrong please try again");

break;

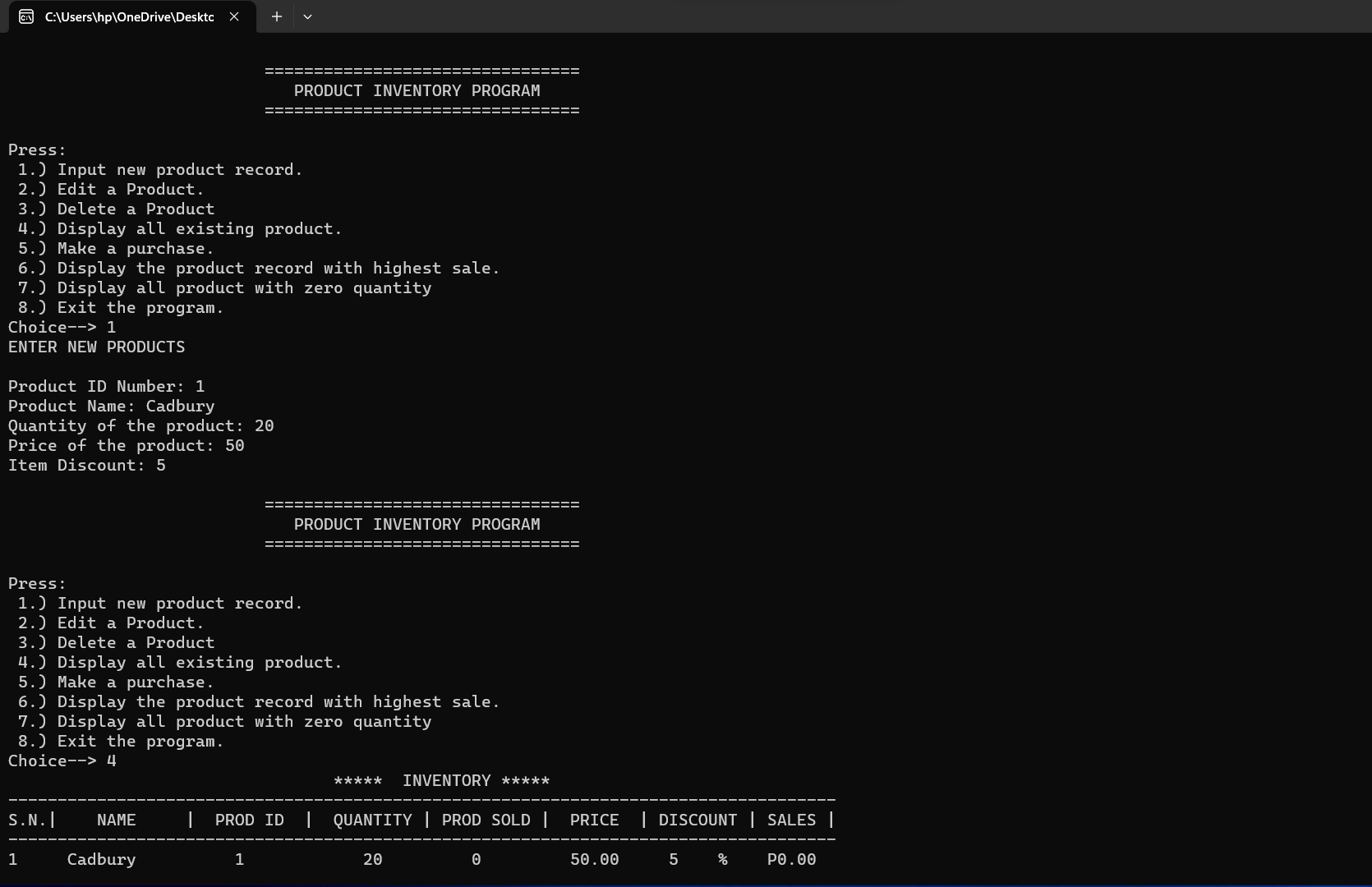
}

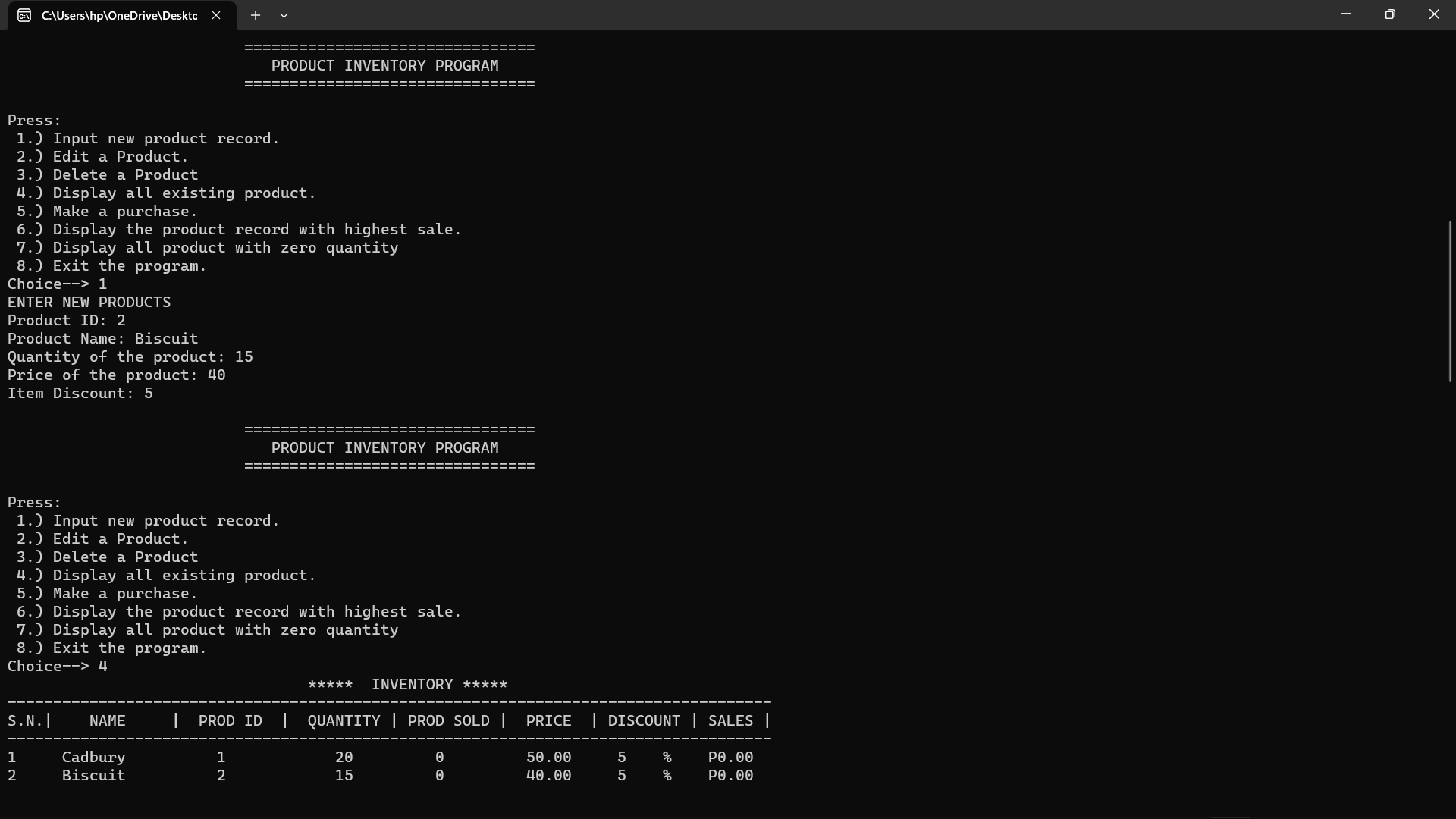
}while(choice!=8); // infinite loop until the user will choose number8 .

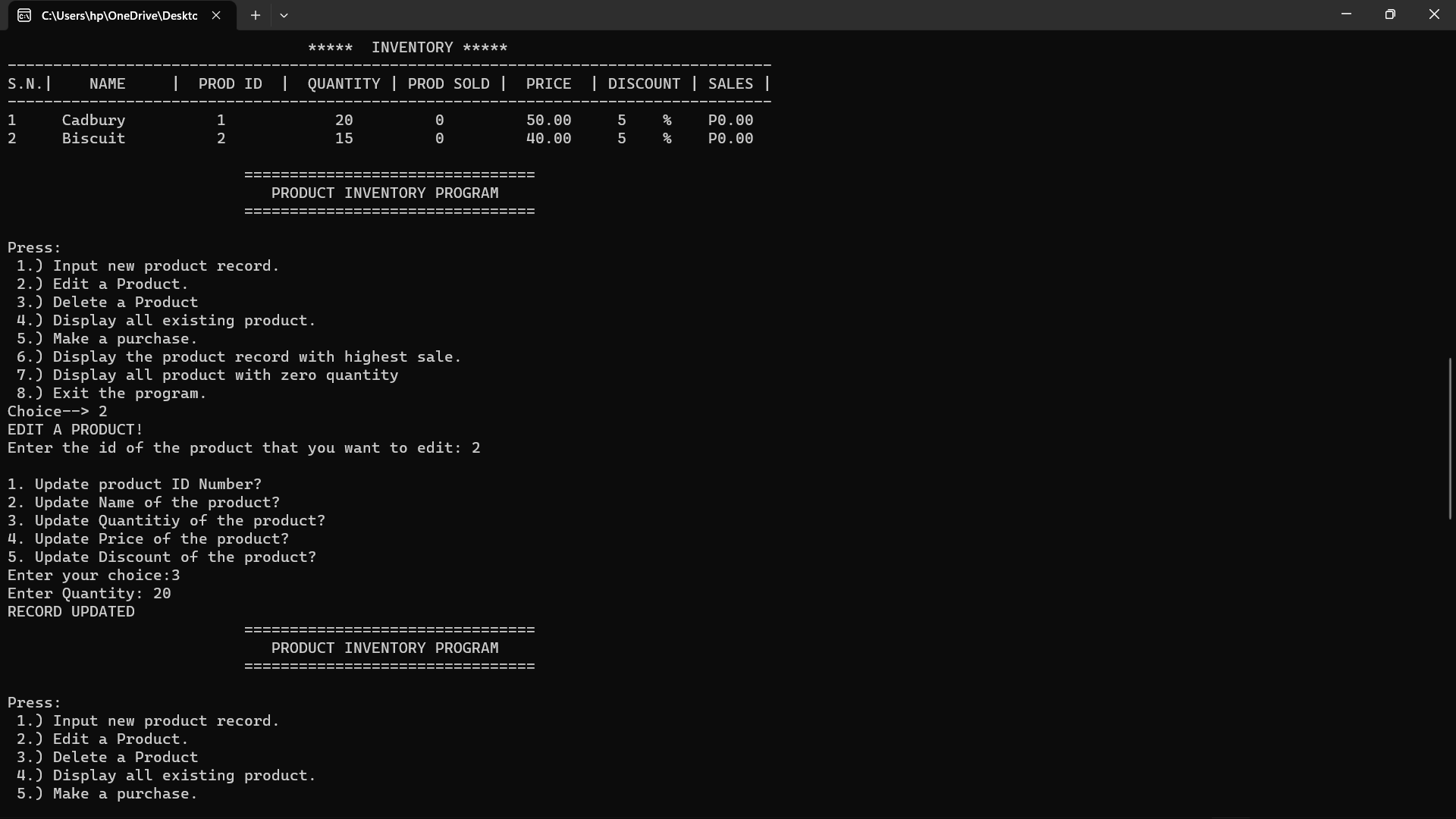
printf("Thankyou for using this program");

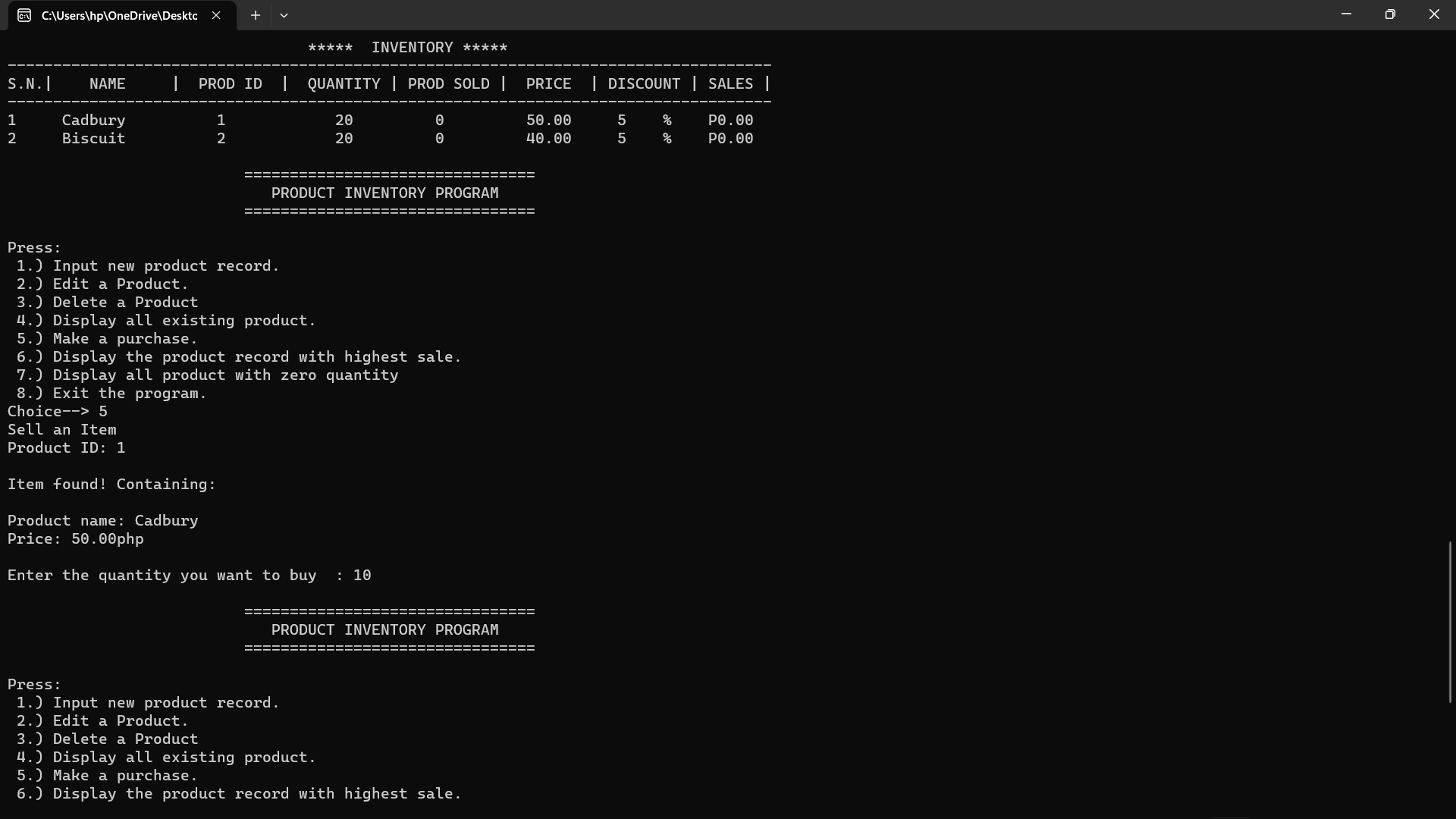
}

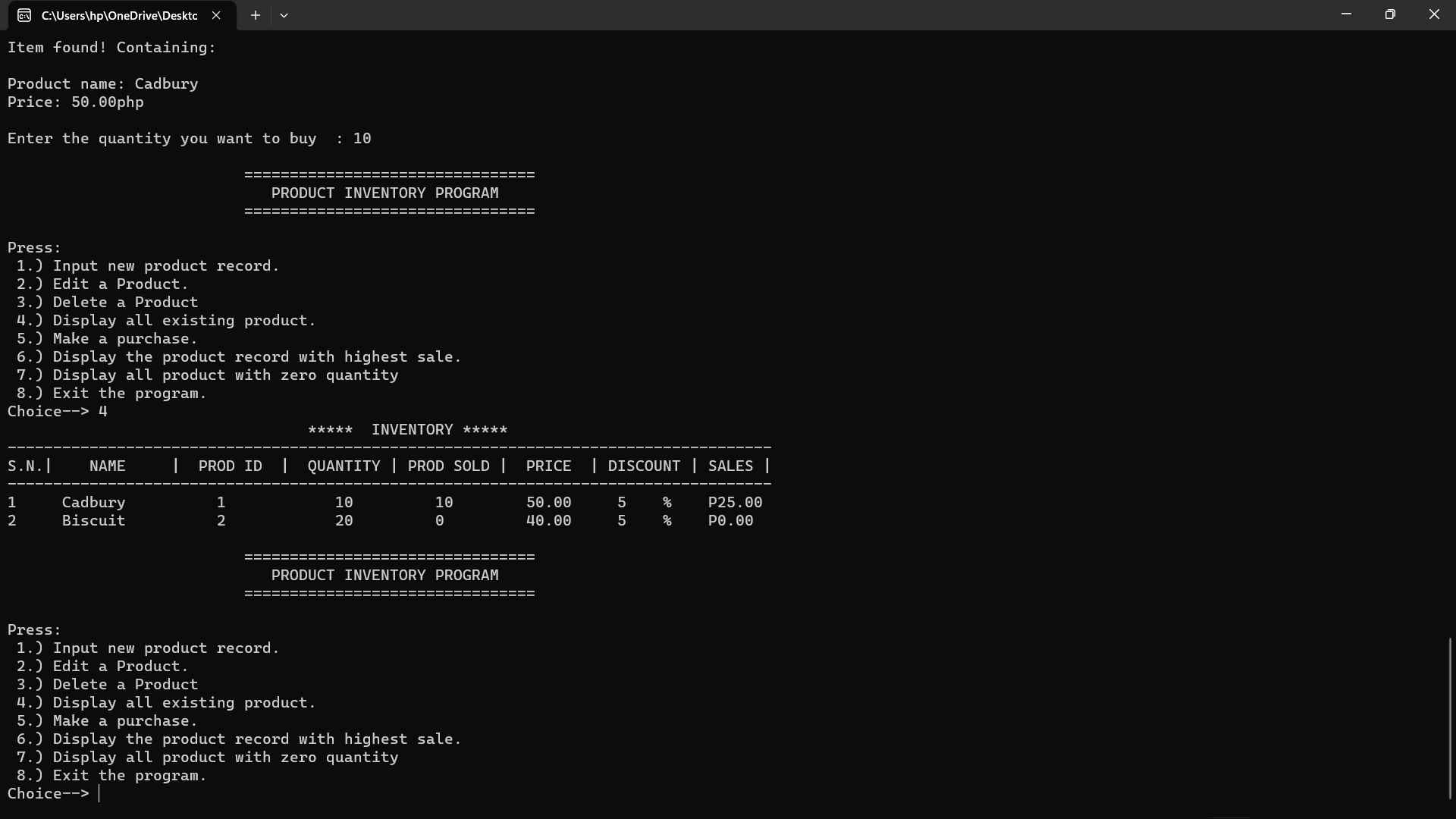
**PROGRAM OUTPUT:**











**FUNCTIONS**:

**struct product( ) -** It is a structure which stores id , name , quantity , numsold , price , discount and sales.

**int readfile( ) –** It reads the values from the file.

**void disZeroQuant( ) –** It calls the function and print the highest product.

**void dispHsale( ) –** It displays the product with highest sale.

**void purchaseprod( ) –** It is the function function for purchasing a product and so displays if the product is not available.

**void addProd( ) –** This function adds product to the file.

**int IDChecker(int i, int j) –** This function checks the input id.

**void editProd( )** **–** This function edits the product information.

**int checkID(char id[]) –** This functions check if the id is available or not.

**void displayprod( ) –** This function displays the output.

**int main ( ) –** This function takes input from user.

**VARIABLE DESCRIPTION**:

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Data Type** | **Variable Description** |
| id | char | Product code/no. |
| name | char | Product name. |
| quantity | int | Remaining quantity of product. Subtract from the original quantity the quantity purchased. |
| numsold | int | Initially zero, when no purchase yet. |
| price | float | Price of one piece of product. |
| discount | int | Discount for this product. |
| sales | float | Accumulated sales, total sales for this product. |
| i | int | It is a iteration for loop. |
| n | int | It is the limit of iteration. |
| high | int | It stores the highest value. |
| choice | int | It takes choice as input. |
| j | int | It is a iteration for loop. |