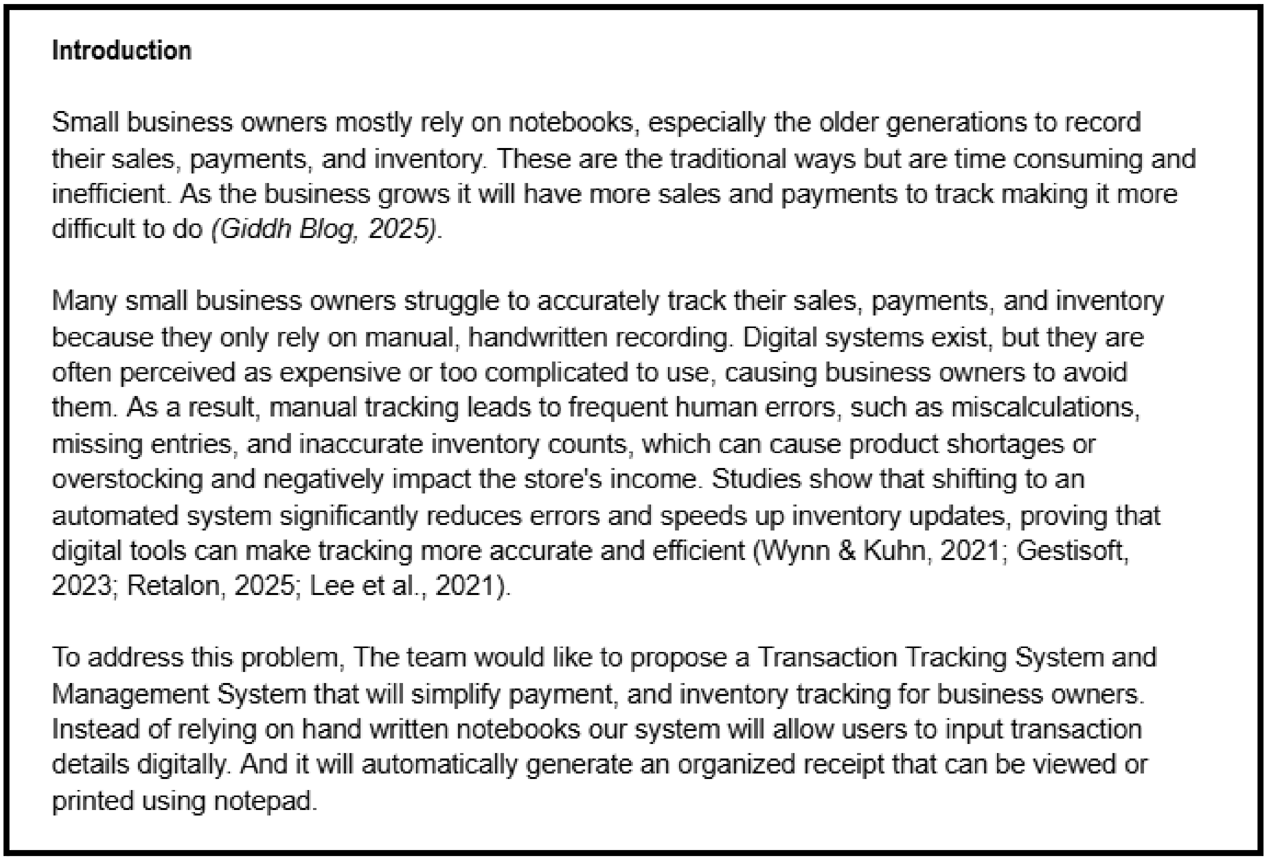
“Transaction Tracking System for Small Business Owners”

Bal, Kevin Ranz J.
Bandigan, Christian Lee C.
Canoy, Hail B.
Pabalan, Hanz Rhayven M.

Work done	Contributions
1. Finalization of Introduction	ALL
2. Finalization of Pseudocode	Canoy
3. Finalization of Code	Bandigan, Pabalan
4. All most done in flowchart	Bal
5. Design of the Code	ALL

Finalization of Introduction:



Introduction

Small business owners mostly rely on notebooks, especially the older generations to record their sales, payments, and inventory. These are the traditional ways but are time consuming and inefficient. As the business grows it will have more sales and payments to track making it more difficult to do *(Giddh Blog, 2025)*.

Many small business owners struggle to accurately track their sales, payments, and inventory because they only rely on manual, handwritten recording. Digital systems exist, but they are often perceived as expensive or too complicated to use, causing business owners to avoid them. As a result, manual tracking leads to frequent human errors, such as miscalculations, missing entries, and inaccurate inventory counts, which can cause product shortages or overstocking and negatively impact the store's income. Studies show that shifting to an automated system significantly reduces errors and speeds up inventory updates, proving that digital tools can make tracking more accurate and efficient (Wynn & Kuhn, 2021; Gestisoft, 2023; Retalon, 2025; Lee et al., 2021).

To address this problem, The team would like to propose a Transaction Tracking System and Management System that will simplify payment, and inventory tracking for business owners. Instead of relying on hand written notebooks our system will allow users to input transaction details digitally. And it will automatically generate an organized receipt that can be viewed or printed using notepad.

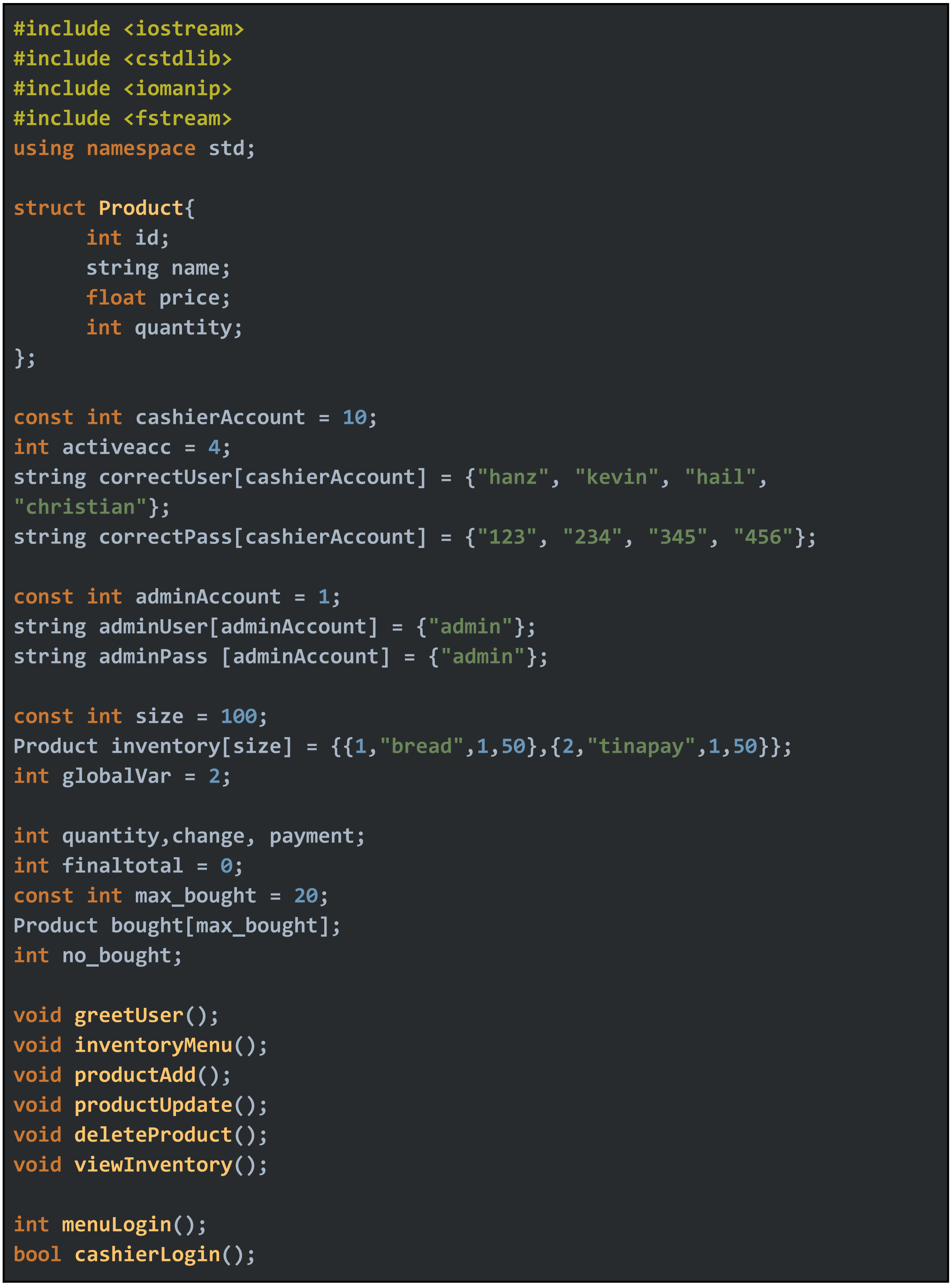
Finalization of Pseudocode:

```
START PROGRAM

SET correctUsername = [ “hanz”, “kevin”, “hail”,  “christian” ]
SET correctPassword = {“123”,”234”,”345”,”456”}
SET adminUsername = “admin”
SET adminPassword = “admin”
SET inventoryCount = 0
SET totalProducts = 0

Login:
OUTPUT “- - - - - Welcome to Transaction Tracking and
Management System - - - - -”
OUTPUT "- - - - - LOGIN - - - - -"
OUTPUT “1. Admin”
OUTPUT “2. Cashier”
OUTPUT “3. Exit”
OUTPUT “Enter Choice:”
INPUT choice

IF choice = 1 THEN
    OUTPUT “Enter Admin Username: “
    INPUT correctAdminusername
    OUTPUT “Enter Admin Password: ”
    INPUT correctAdminpassword
    IF correctAdminusername = adminUsername AND
correctAdminpassword = adminPassword THEN
        GOTO AdminMenu
    ELSE
        OUTPUT “Invalid username or password”
    END IF


ELSE IF choice = 2 THEN
    OUTPUT “Enter Cashier username: “
    INPUT cashierUsername
    OUTPUT “Enter Cashier password:”
    INPUT cashierPassword
    IF cashierUsername = correctUsername AND cashierPassword =
correctPassword THEN
        GOTO CashierMenu
    ELSE
        OUTPUT “Invalid username or password”
    END IF


ELSE
```

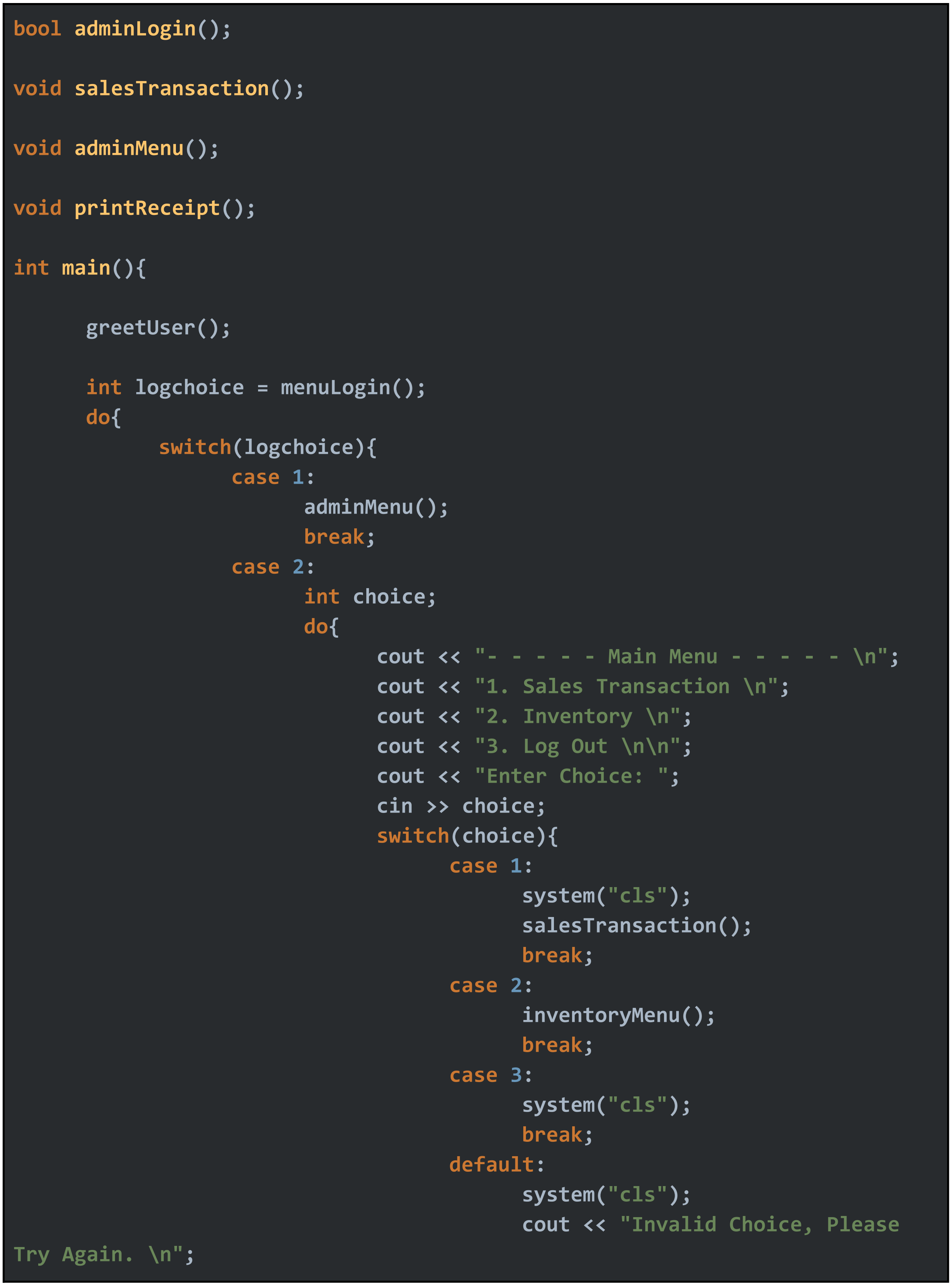
```
    OUTPUT “Exiting program…”
END IF

AdminMenu:
REPEAT
    OUTPUT “- - - - - Admin Menu - - - - -”
    OUTPUT "1. “Add Cashier Accounts"
    OUTPUT "2. “Remove Cashier Accounts"
    OUTPUT "3. “View ALL Cashier Accounts"
    OUTPUT "4. “Log Out"
    OUTPUT “Enter Choice:”
    INPUT adminChoice

    IF adminChoice = 1 THEN
        OUTPUT “Enter Cashier Username: ”
        INPUT cashierUsername
        OUTPUT “Enter Password: ”
        INPUT cashierPassword


    ELSE IF
        adminChoice = 2 THEN
        OUTPUT “Enter Cashier Username: ”
        INPUT cashier
        REMOVE cashier from correctUsername
        OUTPUT “Deleted Successfully”


    ELSE IF adminChoice = 3 THEN
        DISPLAY all correctUsername


    END IF
UNTIL adminChoice = 4
GOTO Login

CashierMenu:
REPEAT
    OUTPUT ”- - - - - Main Menu - - - - -”
    OUTPUT ”1. Sales Transaction”
    OUTPUT ”2. Inventory”
    OUTPUT ”3. Log Out”
    OUTPUT “Enter Choice:”
    INPUT cashierChoice

 IF cashierChoice = 1 THEN
```

```
    REPEAT
        OUTPUT “Enter Product ID: ”
        INPUT productId
        IF productId found THEN
            OUTPUT “Enter Quantity:”
            INPUT quantity
            IF quantity <= stock THEN
                total = total + price * quantity
                Stock = stock - quantity
        ELSE
            OUTPUT "Insufficient Stocks.”
        END IF
    ELSE
            OUTPUT “Product not found”
       END IF
       OUTPUT “Add more Items? (y/n):”
        INPUT more
    UNTIL more = “n”
    OUTPUT "Total Amount: ”
    OUTPUT “Enter Payment:”
    INPUT payment
    IF payment >= total THEN
        OUTPUT “Change: ”, total - payment
        OUTPUT “Transaction complete!!”
    ELSE
        OUTPUT "Insufficient payment!”
    END IF


ELSE IF adminChoice = 2 THEN
    GOTO InventoryMenu
END IF
UNTIL cashierChoice = 3
GOTO Login

InventoryMenu:
REPEAT
    OUTPUT ”- - - - - Inventory - - - - -”
    OUTPUT "1. Add a Product"
    OUTPUT "2. Update a Product"
    OUTPUT "3. Delete a Product"
    OUTPUT "4. View Inventory"
    OUTPUT "5.  Exit”
    OUTPUT “Enter Choice:“
    INPUT choice
    IF choice = 1 THEN
        OUTPUT ”Enter Product ID:”
        INPUT id
        OUTPUT ”Enter product name:”
        INPUT name
        OUTPUT ”Enter price:”
        INPUT price
        OUTPUT ”Enter quantity:”
        INPUT quantity
        ADD to inventory
        OUTPUT "Successfully added product!!"


    ELSE IF choice = 2 THEN
        OUTPUT “Enter Product ID:”
        INPUT productId
        OUTPUT ”Enter New Product Name:”
        INPUT newName
        OUTPUT ”Enter New Price:”
        INPUT newPrice
        OUTPUT ”Enter New Quantity:”
        INPUT newQuantity
        OUTPUT "Successfully updated the product!!"


    ELSE IF choice = 3 THEN
        OUTPUT “Enter Product ID:”
        INPUT productId
        OUTPUT "Product Deleted Successfully"


    ELSE IF choice = 4 THEN
        DISPLAY  all Inventory products
    ELSE
        OUTPUT”Invalid choice”
UNTIL choice = 5
GOTO CashierMenu

OUTPUT “Exiting program…”

END PROGRAM
```

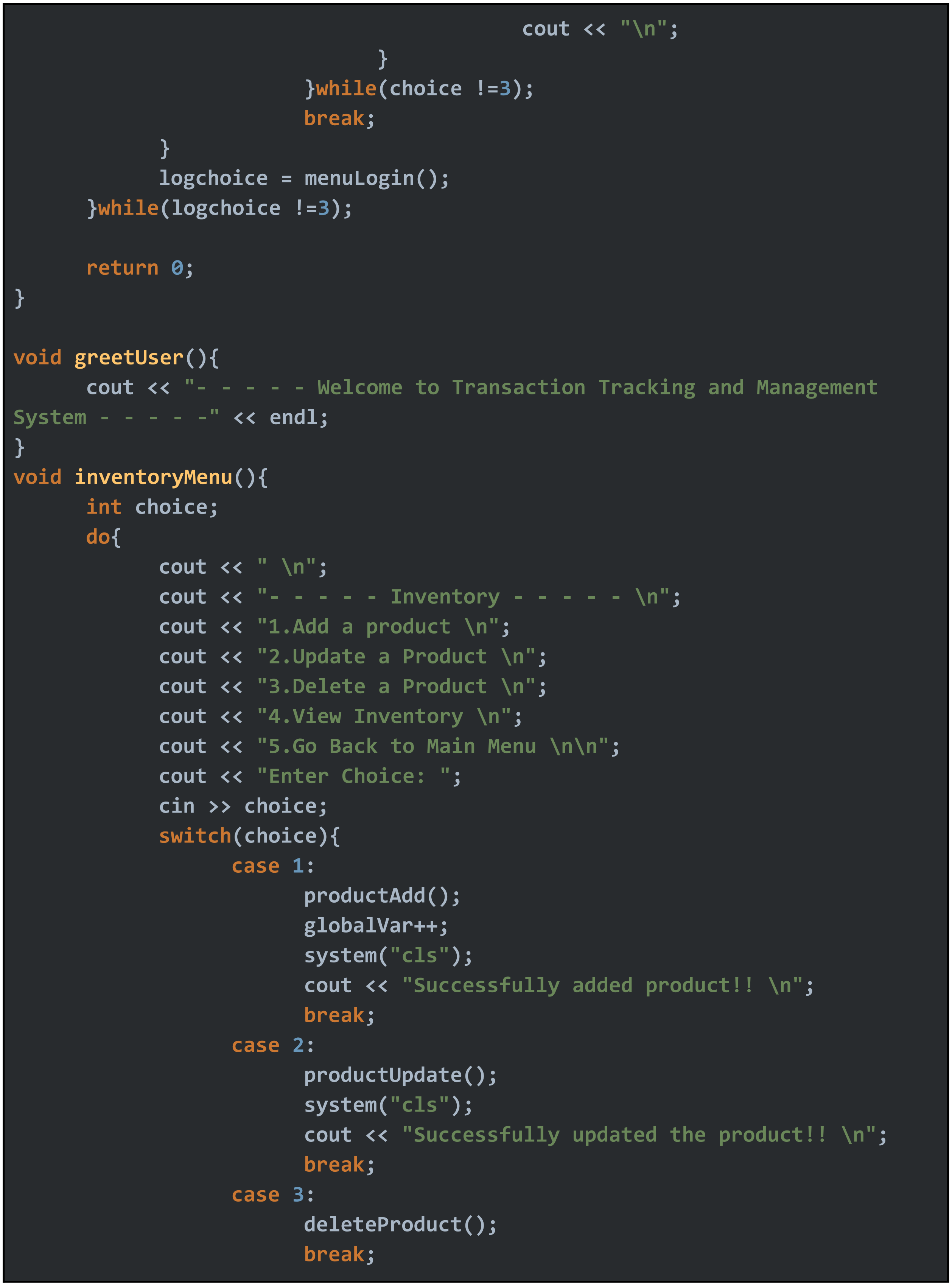
Finalization of Code:



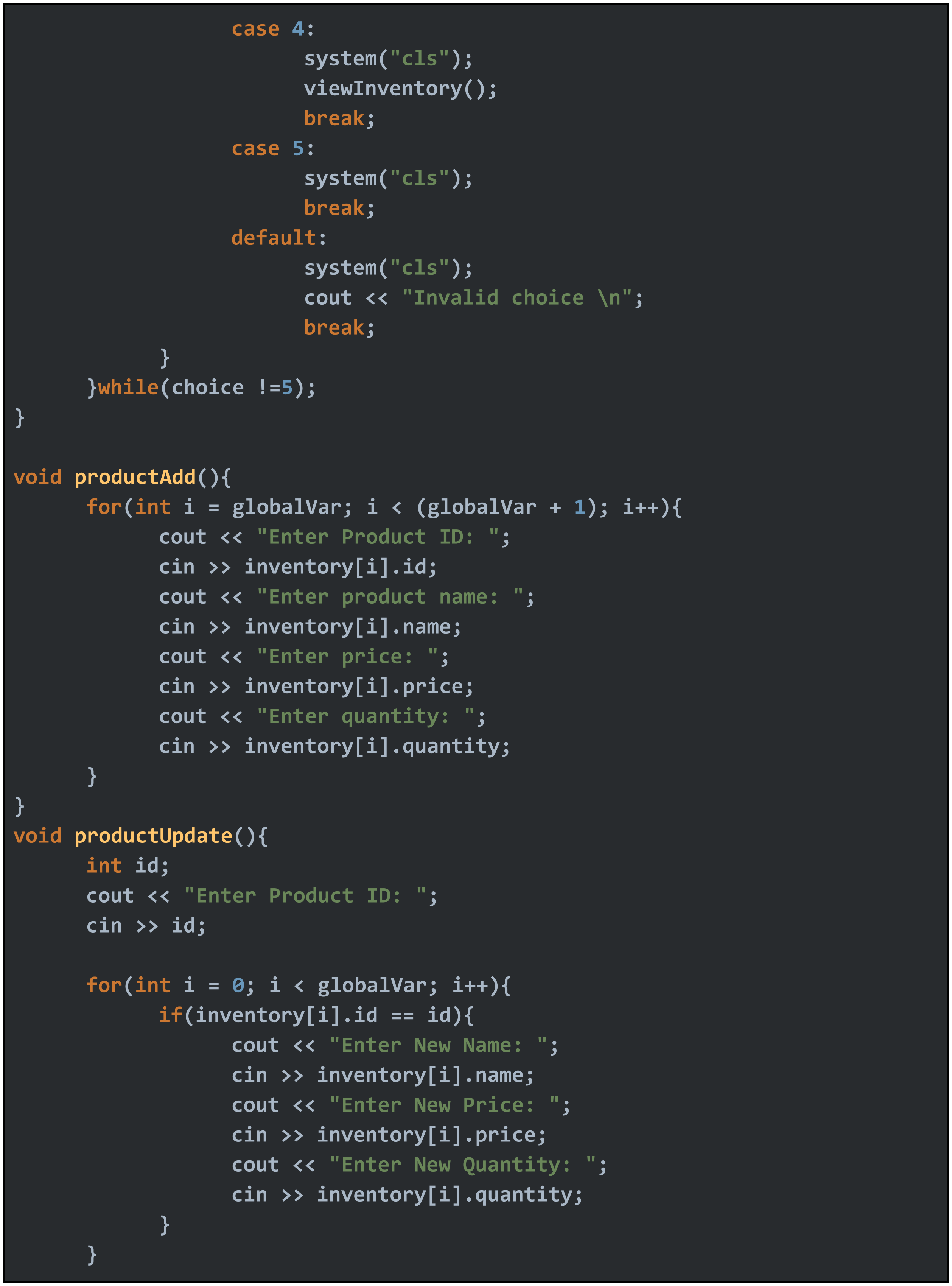
```cpp
#include <iostream>
#include <cstdlib>
#include <iomanip>
#include <fstream>
using namespace std;

struct Product{
	int id;
	string name;
	float price;
	int quantity;
};

const int cashierAccount = 10;
int activeacc = 4;
string correctUser[cashierAccount] = {"hanz", "kevin", "hail",
"christian"};
string correctPass[cashierAccount] = {"123", "234", "345", "456"};

const int adminAccount = 1;
string adminUser[adminAccount] = {"admin"};
string adminPass [adminAccount] = {"admin"};

const int size = 100;
Product inventory[size] = {{1,"bread",1,50},{2,"tinapay",1,50}};
int globalVar = 2;

int quantity,change, payment;
int finaltotal = 0;
const int max_bought = 20;
Product bought[max_bought];
int no_bought;

void greetUser();
void inventoryMenu();
void productAdd();
void productUpdate();
void deleteProduct();
void viewInventory();

int menuLogin();
bool cashierLogin();
```



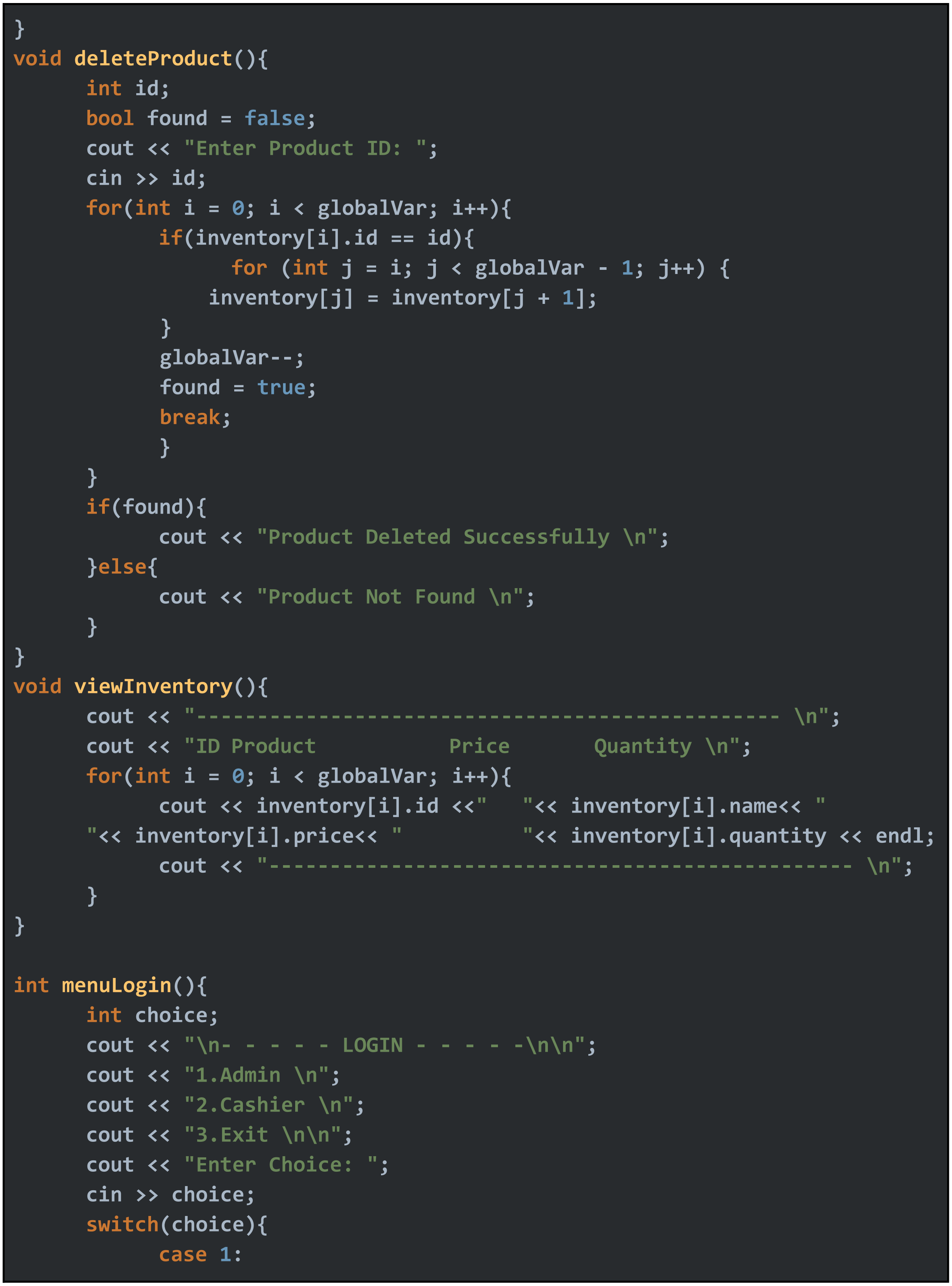
```
bool adminLogin();

void salesTransaction();

void adminMenu();

void printReceipt();

int main(){

      greetUser();

      int logchoice = menuLogin();
      do{
            switch(logchoice){
                  case 1:
                        adminMenu();
                        break;
                  case 2:
                        int choice;
                        do{
                              cout << "- - - - - Main Menu - - - - - \n";
                              cout << "1. Sales Transaction \n";
                              cout << "2. Inventory \n";
                              cout << "3. Log Out \n\n";
                              cout << "Enter Choice: ";
                              cin >> choice;
                              switch(choice){
                                    case 1:
                                          system("cls");
                                          salesTransaction();
                                          break;
                                    case 2:
                                          inventoryMenu();
                                          break;
                                    case 3:
                                          system("cls");
                                          break;
                                    default:
                                          system("cls");
                                          cout << "Invalid Choice, Please
Try Again. \n";
```



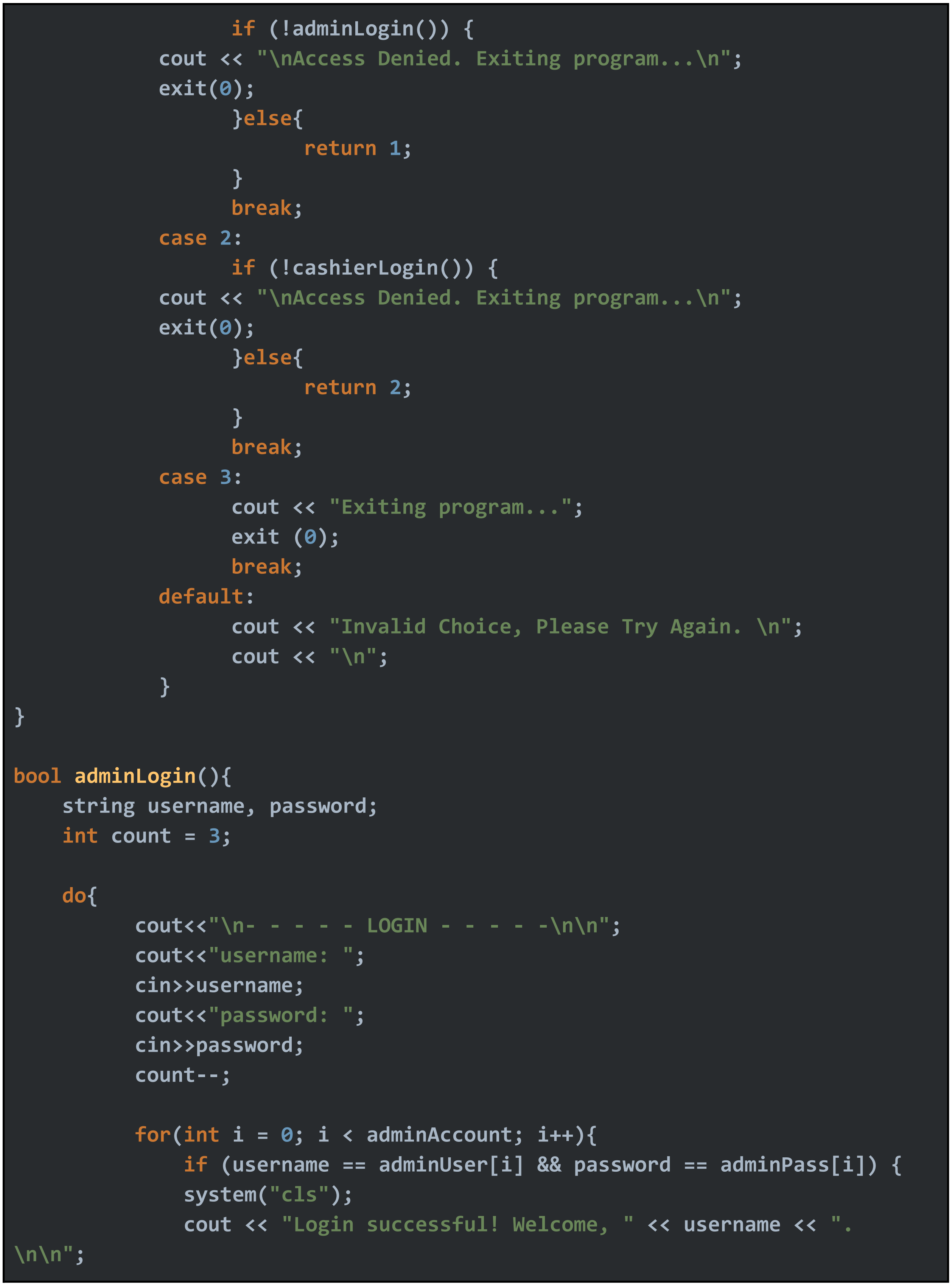
```
                            cout << "\n";
                        }
                    }while(choice !=3);
                    break;
            }
            logchoice = menuLogin();
    }while(logchoice !=3);

    return 0;
}

void greetUser(){
    cout << "- - - - - Welcome to Transaction Tracking and Management
System - - - - -" << endl;
}
void inventoryMenu(){
    int choice;
    do{
        cout << " \n";
        cout << "- - - - - Inventory - - - - - \n";
        cout << "1.Add a product \n";
        cout << "2.Update a Product \n";
        cout << "3.Delete a Product \n";
        cout << "4.View Inventory \n";
        cout << "5.Go Back to Main Menu \n\n";
        cout << "Enter Choice: ";
        cin >> choice;
        switch(choice){
            case 1:
                productAdd();
                globalVar++;
                system("cls");
                cout << "Successfully added product!! \n";
                break;
            case 2:
                productUpdate();
                system("cls");
                cout << "Successfully updated the product!! \n";
                break;
            case 3:
                deleteProduct();
                break;
```



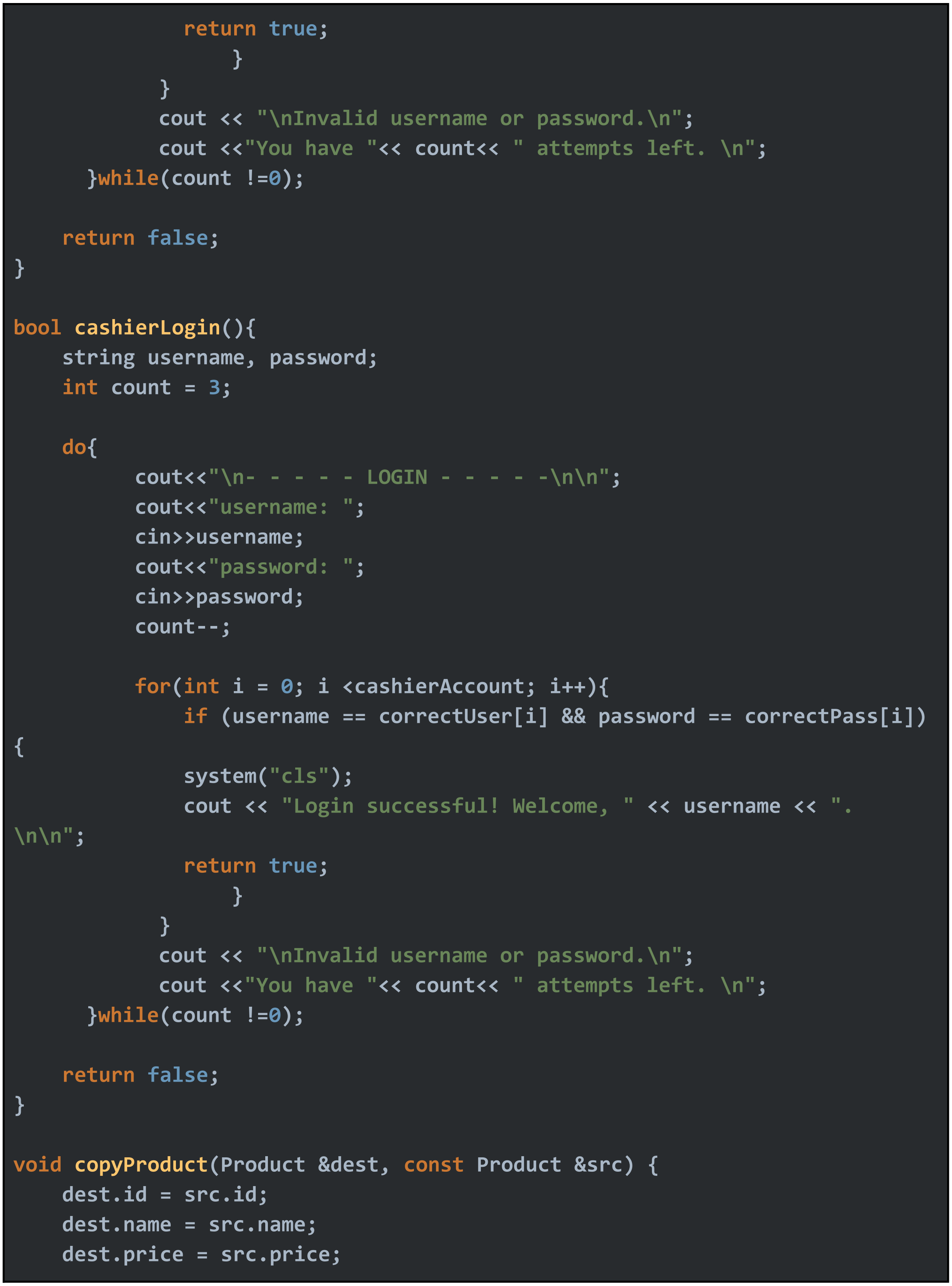
```
            case 4:
                system("cls");
                viewInventory();
                break;
            case 5:
                system("cls");
                break;
            default:
                system("cls");
                cout << "Invalid choice \n";
                break;
        }
    }while(choice !=5);
}

void productAdd(){
    for(int i = globalVar; i < (globalVar + 1); i++){
        cout << "Enter Product ID: ";
        cin >> inventory[i].id;
        cout << "Enter product name: ";
        cin >> inventory[i].name;
        cout << "Enter price: ";
        cin >> inventory[i].price;
        cout << "Enter quantity: ";
        cin >> inventory[i].quantity;
    }
}
void productUpdate(){
    int id;
    cout << "Enter Product ID: ";
    cin >> id;

    for(int i = 0; i < globalVar; i++){
        if(inventory[i].id == id){
            cout << "Enter New Name: ";
            cin >> inventory[i].name;
            cout << "Enter New Price: ";
            cin >> inventory[i].price;
            cout << "Enter New Quantity: ";
            cin >> inventory[i].quantity;
        }
    }
```



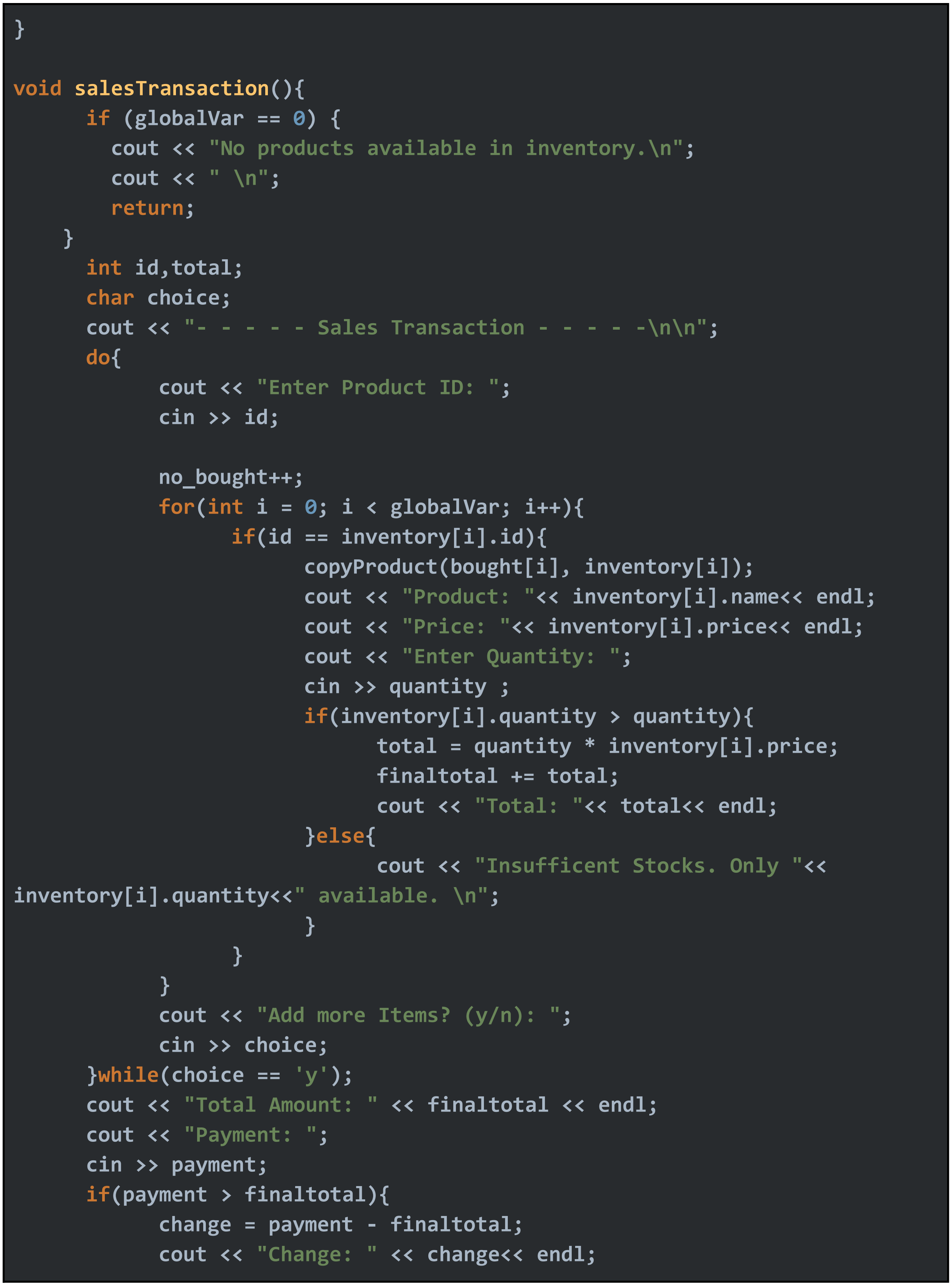
```cpp
}
void deleteProduct(){
      int id;
      bool found = false;
      cout << "Enter Product ID: ";
      cin >> id;
      for(int i = 0; i < globalVar; i++){
            if(inventory[i].id == id){
                  for (int j = i; j < globalVar - 1; j++) {
                inventory[j] = inventory[j + 1];
            }
            globalVar--;
            found = true;
            break;
            }
      }
      if(found){
            cout << "Product Deleted Successfully \n";
      }else{
            cout << "Product Not Found \n";
      }
}
void viewInventory(){
      cout << "------------------------------------------------ \n";
      cout << "ID Product           Price       Quantity \n";
      for(int i = 0; i < globalVar; i++){
            cout << inventory[i].id <<"   "<< inventory[i].name<< "
      "<< inventory[i].price<< "          "<< inventory[i].quantity << endl;
            cout << "------------------------------------------------ \n";
      }
}

int menuLogin(){
      int choice;
      cout << "\n- - - - - LOGIN - - - - -\n\n";
      cout << "1.Admin \n";
      cout << "2.Cashier \n";
      cout << "3.Exit \n\n";
      cout << "Enter Choice: ";
      cin >> choice;
      switch(choice){
            case 1:
```



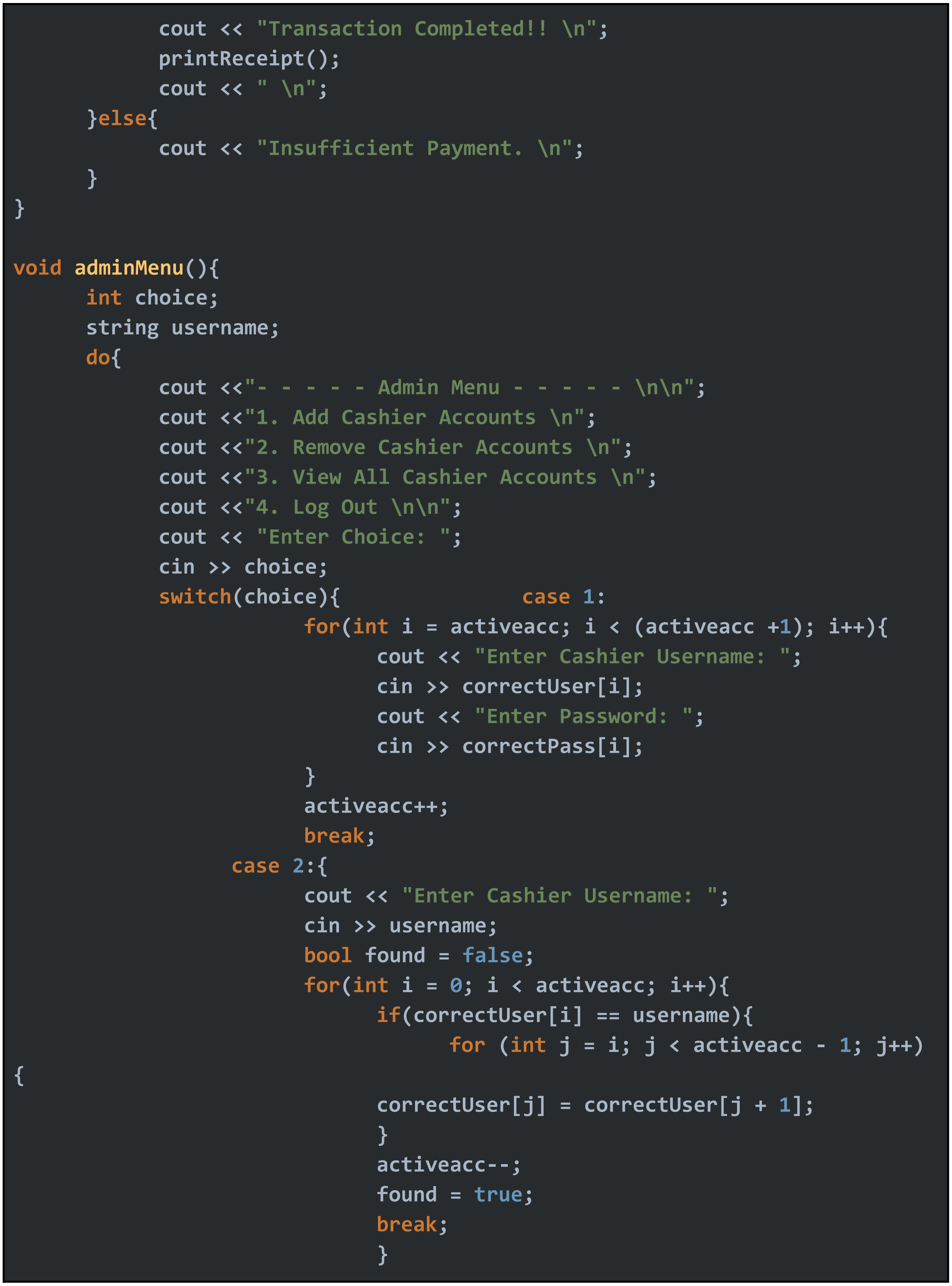
```
                  if (!adminLogin()) {
            cout << "\nAccess Denied. Exiting program...\n";
            exit(0);
                  }else{
                        return 1;
                  }
                  break;
            case 2:
                  if (!cashierLogin()) {
            cout << "\nAccess Denied. Exiting program...\n";
            exit(0);
                  }else{
                        return 2;
                  }
                  break;
            case 3:
                  cout << "Exiting program...";
                  exit (0);
                  break;
            default:
                  cout << "Invalid Choice, Please Try Again. \n";
                  cout << "\n";
            }
}

bool adminLogin(){
    string username, password;
    int count = 3;

    do{
          cout<<"\n- - - - - LOGIN - - - - -\n\n";
          cout<<"username: ";
          cin>>username;
          cout<<"password: ";
          cin>>password;
          count--;

          for(int i = 0; i < adminAccount; i++){
              if (username == adminUser[i] && password == adminPass[i]) {
              system("cls");
              cout << "Login successful! Welcome, " << username << ".
\n\n";
```



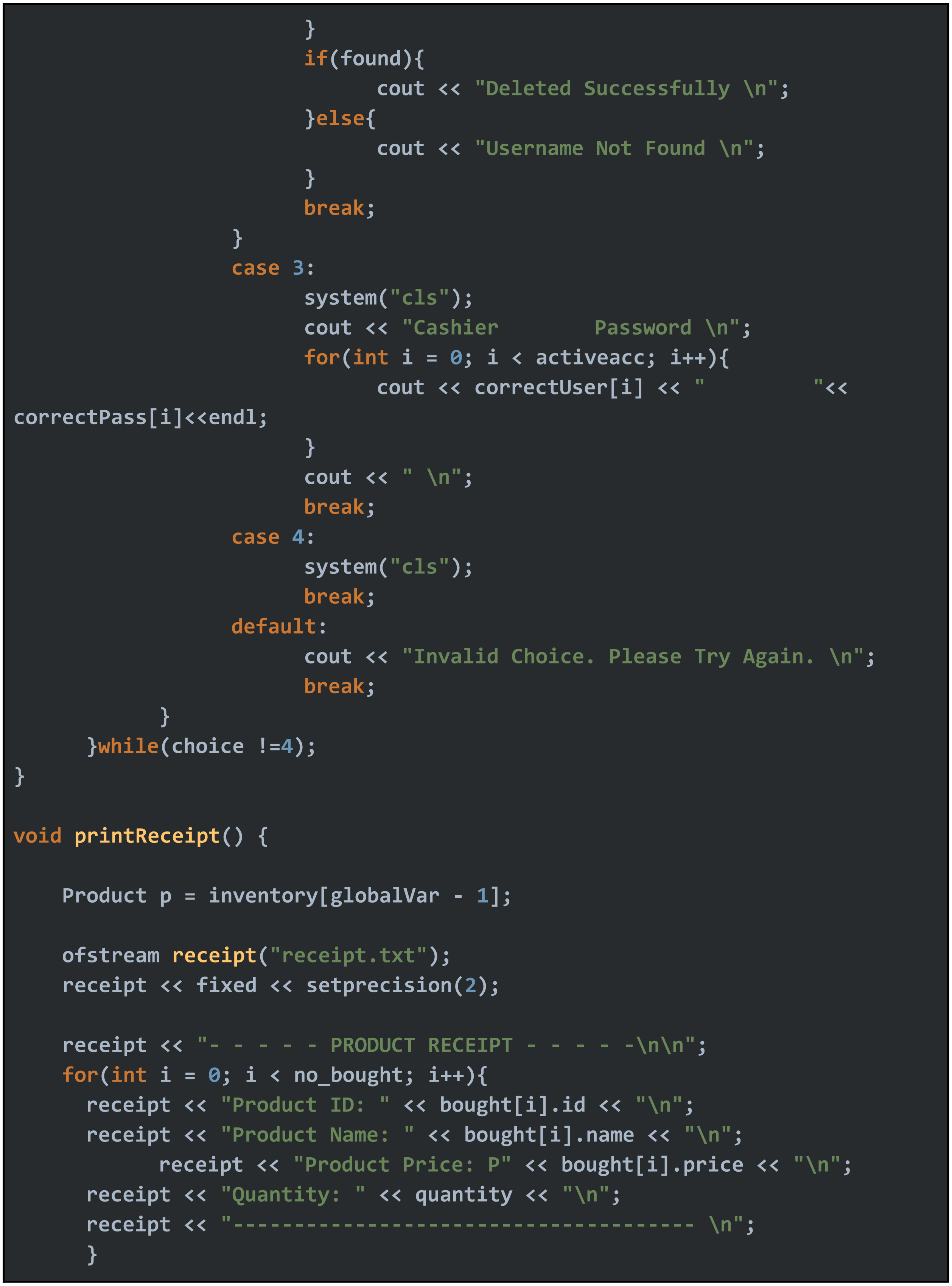
```
              return true;
                  }
            }
            cout << "\nInvalid username or password.\n";
            cout <<"You have "<< count<< " attempts left. \n";
      }while(count !=0);

    return false;
}

bool cashierLogin(){
    string username, password;
    int count = 3;

    do{
          cout<<"\n- - - - - LOGIN - - - - -\n\n";
          cout<<"username: ";
          cin>>username;
          cout<<"password: ";
          cin>>password;
          count--;

          for(int i = 0; i <cashierAccount; i++){
              if (username == correctUser[i] && password == correctPass[i])
{
              system("cls");
              cout << "Login successful! Welcome, " << username << ".
\n\n";
              return true;
                  }
            }
            cout << "\nInvalid username or password.\n";
            cout <<"You have "<< count<< " attempts left. \n";
      }while(count !=0);

    return false;
}

void copyProduct(Product &dest, const Product &src) {
    dest.id = src.id;
    dest.name = src.name;
    dest.price = src.price;
```



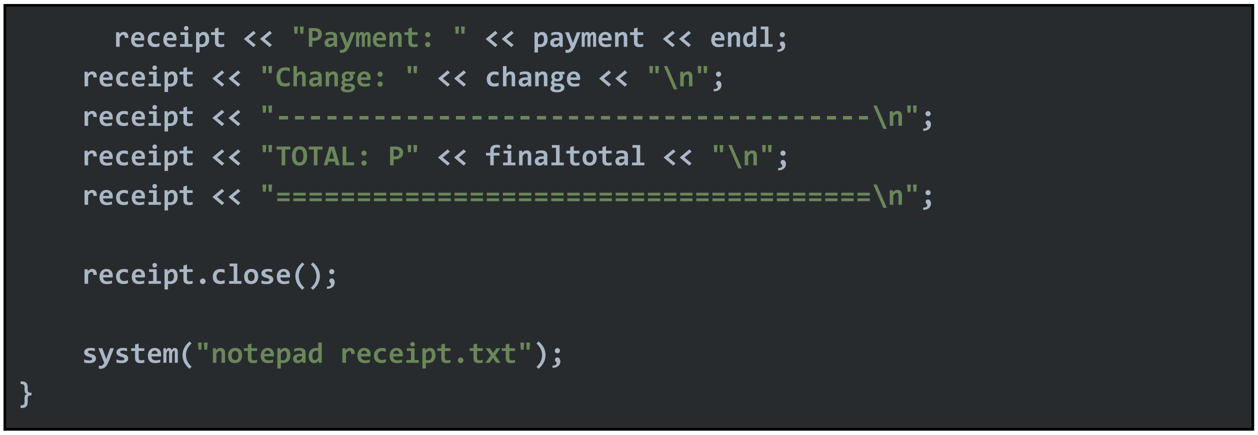
```
}

void salesTransaction(){
      if (globalVar == 0) {
        cout << "No products available in inventory.\n";
        cout << " \n";
        return;
    }
      int id,total;
      char choice;
      cout << "- - - - - Sales Transaction - - - - -\n\n";
      do{
            cout << "Enter Product ID: ";
            cin >> id;

            no_bought++;
            for(int i = 0; i < globalVar; i++){
                  if(id == inventory[i].id){
                        copyProduct(bought[i], inventory[i]);
                        cout << "Product: "<< inventory[i].name<< endl;
                        cout << "Price: "<< inventory[i].price<< endl;
                        cout << "Enter Quantity: ";
                        cin >> quantity ;
                        if(inventory[i].quantity > quantity){
                              total = quantity * inventory[i].price;
                              finaltotal += total;
                              cout << "Total: "<< total<< endl;
                        }else{
                              cout << "Insufficent Stocks. Only "<<
inventory[i].quantity<<" available. \n";
                        }
                  }
            }
            cout << "Add more Items? (y/n): ";
            cin >> choice;
      }while(choice == 'y');
      cout << "Total Amount: " << finaltotal << endl;
      cout << "Payment: ";
      cin >> payment;
      if(payment > finaltotal){
            change = payment - finaltotal;
            cout << "Change: " << change<< endl;
```



```
        cout << "Transaction Completed!! \n";
        printReceipt();
        cout << " \n";
    }else{
        cout << "Insufficient Payment. \n";
    }
}

void adminMenu(){
    int choice;
    string username;
    do{
        cout <<"- - - - - Admin Menu - - - - - \n\n";
        cout <<"1. Add Cashier Accounts \n";
        cout <<"2. Remove Cashier Accounts \n";
        cout <<"3. View All Cashier Accounts \n";
        cout <<"4. Log Out \n\n";
        cout << "Enter Choice: ";
        cin >> choice;
        switch(choice){               case 1:
                for(int i = activeacc; i < (activeacc +1); i++){
                    cout << "Enter Cashier Username: ";
                    cin >> correctUser[i];
                    cout << "Enter Password: ";
                    cin >> correctPass[i];
                }
                activeacc++;
                break;
            case 2:{
                cout << "Enter Cashier Username: ";
                cin >> username;
                bool found = false;
                for(int i = 0; i < activeacc; i++){
                    if(correctUser[i] == username){
                        for (int j = i; j < activeacc - 1; j++)
{
                    correctUser[j] = correctUser[j + 1];
                    }
                    activeacc--;
                    found = true;
                    break;
                    }
```

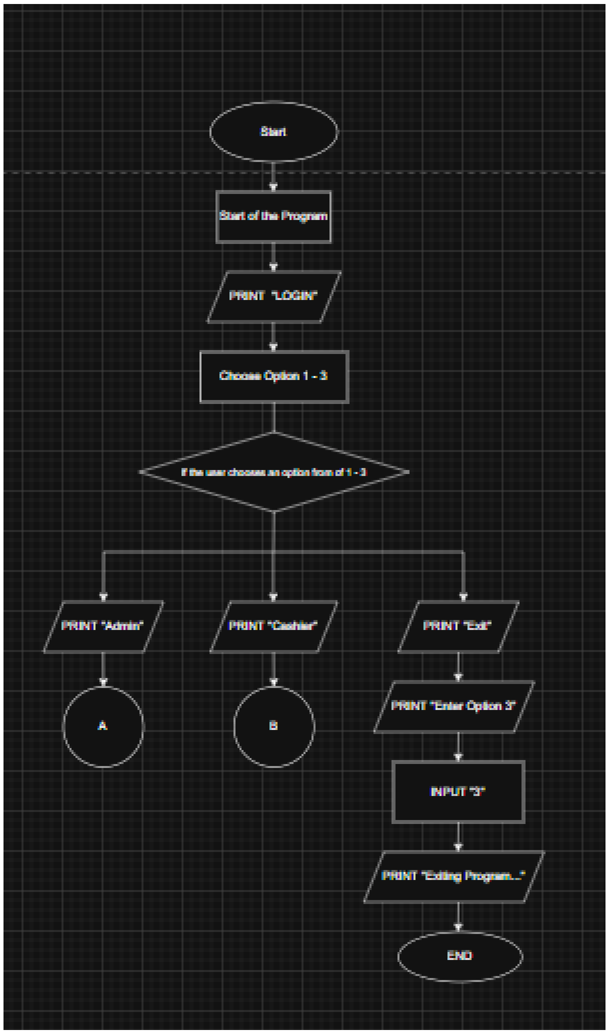


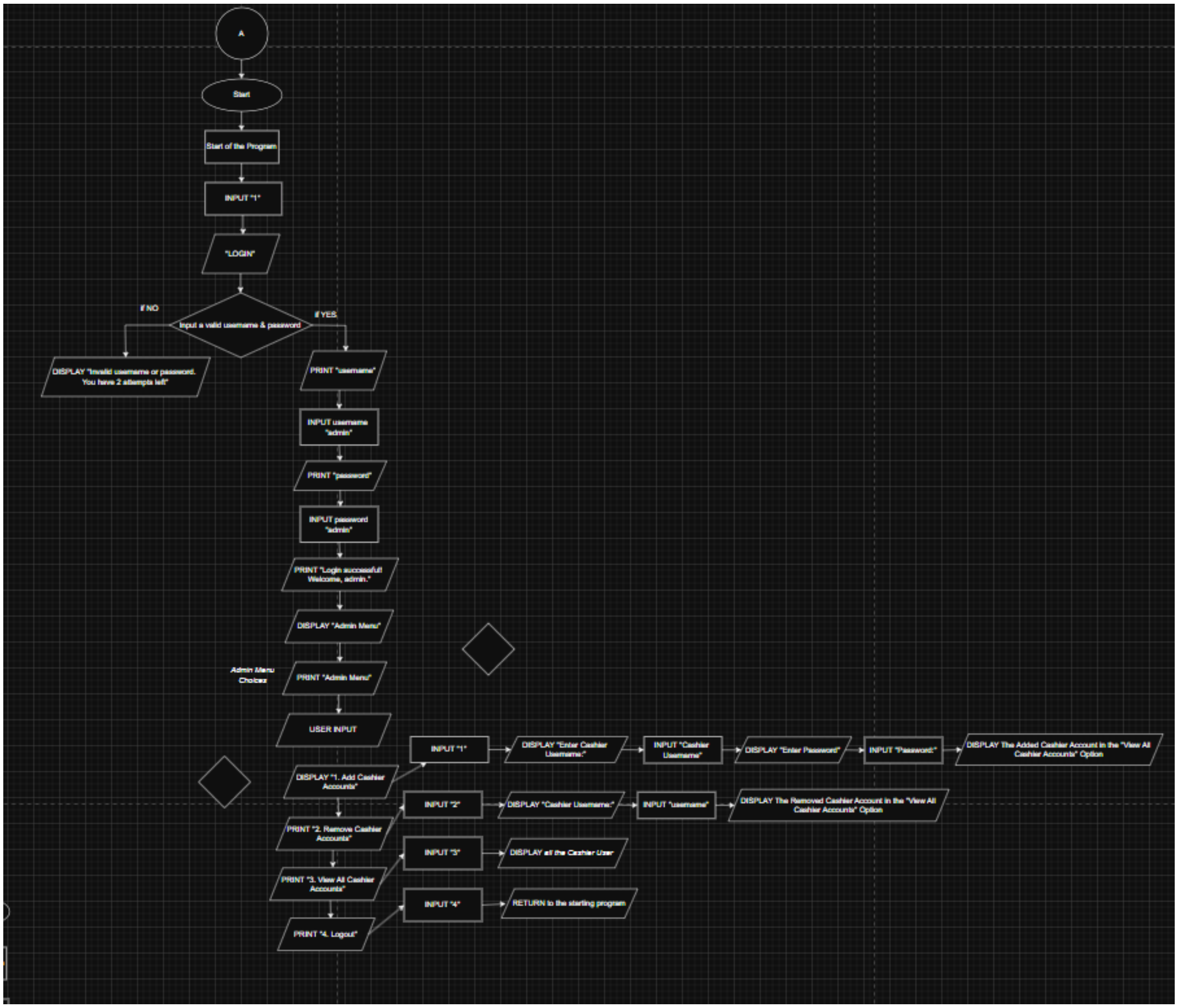
```
                        }
                        if(found){
                              cout << "Deleted Successfully \n";
                        }else{
                              cout << "Username Not Found \n";
                        }
                        break;
                  }
                  case 3:
                        system("cls");
                        cout << "Cashier        Password \n";
                        for(int i = 0; i < activeacc; i++){
                              cout << correctUser[i] << "         "<<
correctPass[i]<<endl;
                        }
                        cout << " \n";
                        break;
                  case 4:
                        system("cls");
                        break;
                  default:
                        cout << "Invalid Choice. Please Try Again. \n";
                        break;
            }
      }while(choice !=4);
}

void printReceipt() {

    Product p = inventory[globalVar - 1];

    ofstream receipt("receipt.txt");
    receipt << fixed << setprecision(2);

    receipt << "- - - - - PRODUCT RECEIPT - - - - -\n\n";
    for(int i = 0; i < no_bought; i++){
      receipt << "Product ID: " << bought[i].id << "\n";
      receipt << "Product Name: " << bought[i].name << "\n";
            receipt << "Product Price: P" << bought[i].price << "\n";
      receipt << "Quantity: " << quantity << "\n";
      receipt << "-------------------------------------- \n";
      }
```

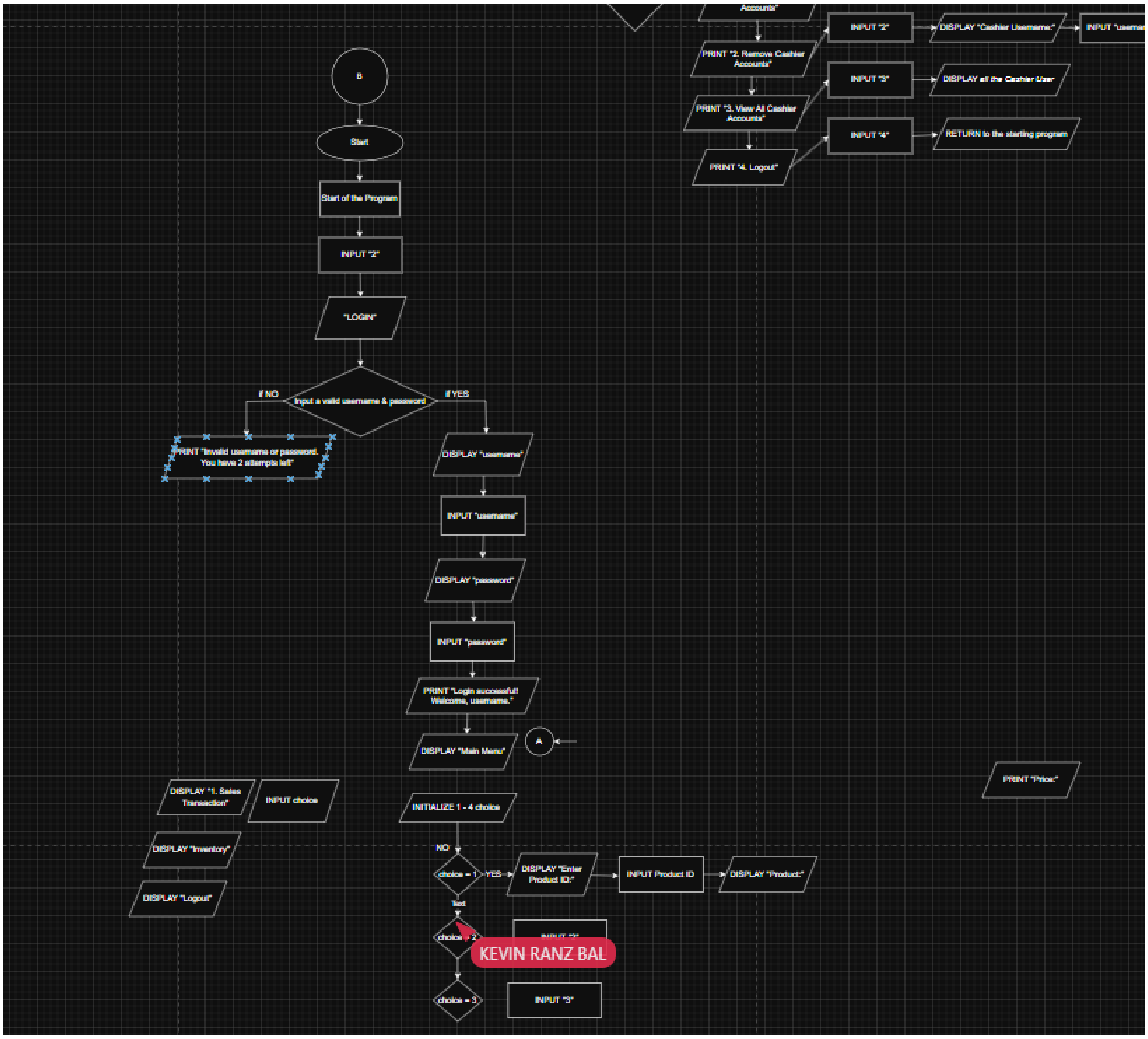


```
      receipt << "Payment: " << payment << endl;
    receipt << "Change: " << change << "\n";
    receipt << "-------------------------------------\n";
    receipt << "TOTAL: P" << finaltotal << "\n";
    receipt << "=====================================\n";

    receipt.close();

    system("notepad receipt.txt");
}
```

Flowchart:


Start
Start of the Program
PRINT "LOGIN"
Choose Option 1 - 3
If the user chooses an option from of 1 - 3
PRINT "Admin"
PRINT "Cashier"
PRINT "Exit"
A
B
PRINT "Enter Option 3"
INPUT "3"
PRINT "Exiting Program..."
END


A
Start
Start of the Program
INPUT "1"
"LOGIN"
if NO
Input a valid username & password
if YES
DISPLAY "Invalid username or password. You have 2 attempts left"
PRINT "username"
INPUT username "admin"
PRINT "password"
INPUT password "admin"
PRINT "Login successful! Welcome, admin."
DISPLAY "Admin Menu"
Admin Menu Choices
PRINT "Admin Menu"
USER INPUT
INPUT "1"
DISPLAY "Enter Cashier Username:"
INPUT "Cashier Username"
DISPLAY "Enter Password"
INPUT "Password:"
DISPLAY The Added Cashier Account in the "View All Cashier Accounts" Option
DISPLAY "1. Add Cashier Accounts"
INPUT "2"
DISPLAY "Cashier Username:"
INPUT "username"
DISPLAY The Removed Cashier Account in the "View All Cashier Accounts" Option
PRINT "2. Remove Cashier Accounts"
INPUT "3"
DISPLAY all the Cashier User
PRINT "3. View All Cashier Accounts"
INPUT "4"
RETURN to the starting program
PRINT "4. Logout"


Accounts"
INPUT "2"
DISPLAY "Cashier Username:"
INPUT "usernam
PRINT "2. Remove Cashier Accounts"
INPUT "3"
DISPLAY all the Cashier User
PRINT "3. View All Cashier Accounts"
INPUT "4"
RETURN to the starting program
PRINT "4. Logout"
B
Start
Start of the Program
INPUT "2"
"LOGIN"
if NO
Input a valid username & password
if YES
PRINT "Invalid username or password. You have 2 attempts left"
DISPLAY "username"
INPUT "username"
DISPLAY "password"
INPUT "password"
PRINT "Login successful! Welcome, username."
DISPLAY "Main Menu"
A
PRINT "Price:"
DISPLAY "1. Sales Transaction"
INPUT choice
INITIALIZE 1 - 4 choice
DISPLAY "Inventory"
DISPLAY "Logout"
NO
choice = 1
YES
DISPLAY "Enter Product ID:"
INPUT Product ID
DISPLAY "Product:"
Text
choice = 2
INPUT "2"
KEVIN RANZ BAL
choice = 3
INPUT "3"