12/21/2022

ATM MACHINE

***SUBMITTING TO:*** MISS RIMSHA JAVED.

***BY***: **MUHAMMAD TAHA KAHN\_FA22-BSCS-0116 & JETHANAND HEMANI\_FA22-BSCS-0100**.

***INTRODUCTION:***

**Automated teller machines**, or **ATMs**, are a need in modern society. It stops us from always carrying around a significant sum of cash and allows the user to withdraw money at any time. In addition to using it to withdraw money, users can also use it to send money. The user must first enter their ID and password in order to access their account. This way, they can also pay their school fees, electricity, gas, and water bills. They can view information about their accounts, such as a balance inquiry or a small statement, among others. This application was created in C++, and the user ID and password were kept in a single file called "**practice2.txt".** This file will only open when the user takes any action involving their User ID and Password, such as changing their pin or logging in.

***PROBLEM STATEMENT:***

Since theft and robbery are growing more frequent in today's environment, it makes sense that people worry about always having their money on them. They can't carry a lot of money around with them at all times, especially when they are travelling or need to transfer money to family members who live far away.

***SOLUTION:***

We have chosen to create an **ATM machine**, also known as an **Automated Teller Machine**, in order to address the aforementioned issue. With the aid of these devices, you may do self-service transactions without going to the bank branch or talking to a real teller. Cash withdrawals, cash deposits, and fund transfers are just a few of the many transactions possible with **Automated Teller Machines**, or **ATMs**.

***FEATURES:***

Many features are included in this project, such as **fast cash**, which enables users to take money right away without entering a specific quantity. **Cash withdrawal** is similar to fast cash, with the exception that the user enters their own desired amount of money in this function. Users can check their account balances using the **balance inquiry** feature. The user may view a **mini-statement** to review their transactional history. If a user wants to modify the password for their account, they can do so by using the **Pin Change** option. Transferring money across accounts is made possible by **Funds Transfer** feature. **Payment** feature allows the User to utilise this tool to pay their utility bills, school fees, and other debts.

***SOURCE CODE:***

#include<conio.h>

#include<iostream>

#include<fstream>

#include<string>

using namespace std;

void Title();

void water();

void gas();

void electricity();

void school();

void payment();

void Payment\_Title();

void fastCash();

void FastCash\_Title();

void withdraw();

void withdraw\_Title();

void login\_page();

void funds\_transfer();

void Balance\_Inquiry();

void Mini\_Statement\_Title();

void mini\_statement();

void Pin\_Change();

int account\_balance=250000,funds=0,account\_number,cash\_withdraw=0;

string mini="Your Mini Statement is: ",mini\_cash1,mini\_cash2,mini\_cash3,mini\_cash4,mini\_cash5,mini\_cash6,mini\_cash7;

string mini\_cash8,mini\_gas,mini\_water,mini\_electric,mini\_city,mini\_allied,mini\_beacon,mini\_smart;

string ID\_1,ID\_2,pass\_2,current\_pass,new\_pass,new\_pass\_2,pass\_1,acc\_balance\_1,acc\_balance\_2,ID;

main()

{

login:

int select;

string pass;

ifstream read("practice2.txt");

getline(read,acc\_balance\_1);

getline(read,ID\_1);

getline(read,pass\_1);

getline(read,acc\_balance\_2);

getline(read,ID\_2);

getline(read,pass\_2);

read.close();

system("cls");

login\_page();

cout<<"\nEnter Your ID: ";

cin>>ID;

if(ID.compare(ID\_2)!=0 && ID.compare(ID\_1)!=0)

{

cout<<"Incorrect ID";

cout<<"\nPress any key to continue";

getch();

goto login;

}

cout<<"\nEnter Your Password: ";

fflush(stdin);

getline(std::cin, pass);

if(ID==ID\_1 && pass==pass\_1 || ID==ID\_2 && pass==pass\_2)

{

main\_menu:

system("cls");

Title();

cout<<"\n1. Fast Cash 2. Cash Withdrawal\n";

cout<<"3. Balance Inquiry 4. Mini Statement\n";

cout<<"5. Pin Change 6. Funds Transfer\n";

cout<<"7. Payments 8. Sign Out\n";

cout<<"Select the operation you want to perform:";

cin>> select;

system("cls");

ifstream read("practice2.txt");

switch (select)

{

case 1:

fastCash();

break;

case 2:

withdraw();

goto main\_menu;

break;

case 3:

Balance\_Inquiry();

cout<<"You currently have "<<account\_balance<<"rs in your account\n";

cout<<"\nPress any key to continue";

getch();

goto main\_menu;

break;

case 4:

system("cls");

Mini\_Statement\_Title();

mini\_statement();

break;

case 5:

login\_change:

system("cls");

Pin\_Change();

cout<<"Enter your current password: ";

getline(read,acc\_balance\_1);

getline(read,ID\_1);

getline(read,pass);

getline(read,acc\_balance\_2);

getline(read,ID\_2);

getline(read,pass\_2);

fflush(stdin);

getline(std::cin,current\_pass);

if(current\_pass==pass\_1 || current\_pass==pass\_2)

{

cout<<"Enter new password: ";

if(ID.compare(ID\_2)==0)

{

fflush(stdin);

getline(std::cin,new\_pass\_2);

}

else

{

fflush(stdin);

getline(std::cin,new\_pass);

}

ofstream write("temp.txt");

ID\_1="Taha123";

ID\_2="Jethanand321";

write<<acc\_balance\_1<<"\n";

write<<ID\_1;

if(ID.compare(ID\_1)==0)

{

write<<"\n"<<new\_pass<<"\n";

}

else

{

write<<"\n"<<pass\_1<<"\n";

}

write<<acc\_balance\_2<<"\n";

write<<ID\_2;

if(ID.compare(ID\_2)==0)

{

write<<"\n"<<new\_pass\_2<<"\n";

}

else

{

write<<"\n"<<pass\_2<<"\n";

}

write.close();

read.close();

remove("practice2.txt");

rename("temp.txt","practice2.txt");

goto login;

}

else

{

cout<<"Incorrect pass";

getch();

goto login\_change;

}

break;

case 6:

funds\_transfer();

goto main\_menu;

break;

case 7:

payment();

break;

case 8:

goto login;

break;

default:

cout<<"Incorrect input\n";

cout<<"\nPress any key to continue";

getch();

break;

}

goto main\_menu;

}

else

{

cout<<"Incorrect Password";

cout<<"\nPress any key to continue";

getch();

system("cls");

goto login;

}

}

void Pin\_Change()

{

cout<<("============================================\n");

cout<<(" PIN CHANGE \n");

cout<<("============================================\n\n");

}

void Funds\_Transfer()

{

cout<<("============================================\n");

cout<<(" FUNDS TRANSSFER \n");

cout<<("============================================\n\n");

}

void Balance\_Inquiry()

{

cout<<("============================================\n");

cout<<(" BALANCE INQUIRY \n");

cout<<("============================================\n\n");

}

void Withdraw\_Title(){

{

cout<<("============================================\n");

cout<<(" CASH WITHDRAWAL \n");

cout<<("============================================\n\n");

}

}

void FastCash\_Title()

{

cout<<("============================================\n");

cout<<(" FAST CASH \n");

cout<<("============================================\n\n");

}

void Payment\_Title()

{

cout<<("============================================\n");

cout<<(" PAYMENT \n");

cout<<("============================================\n\n");

}

void Mini\_Statement\_Title()

{

cout<<("============================================\n");

cout<<(" MINI STATEMENT \n");

cout<<("============================================\n\n");

}

void login\_page()

{

cout<<("============================================\n");

cout<<(" LOGIN PAGE \n");

cout<<("============================================\n\n");

}

void Title()

{

cout<<("============================================\n");

cout<<(" WELCOME TO ATM ! \n");

cout<<("============================================\n");

}

void water()

{

string address;

int bill;

system("cls");

cout<<"\nEnter your house address: ";

cin>>address;

system("cls");

getline(cin,address);

cout<<"\nYour bill is 3000rs";

cout<<"\nDo you want to pay the bill to Water-Board ?";

cout<<"\n1. Yes \n2. No\n";

cin>>bill;

switch(bill)

{

case 1:

system("cls");

if(3000>account\_balance)

{

cout<<"\nNot enough balance";

cout<<"\nPress any key to continue";

getch();

}

else

{

mini\_water="\n==>3000rs were paid to Water-Board\n";

account\_balance=account\_balance-3000;

cout<<"You have successfully paid the bill to Water-Board !\n";

cout<<"Press any key to continue";

getch();

}

break;

case 2:

system("cls");

cout<<"The bill was not paid!\n";

cout<<"Press any key to continue";

getch();

break;

default:

system("cls");

cout<<"Incorrect input\n";

cout<<"\nPress any key to continue";

getch();

break;

}

}

void gas(){

string address;

int bill;

system("cls");

cout<<"\nEnter your house address: ";

cin>>address;

getline(cin,address);

system("cls");

cout<<"\nYour bill is 5000rs";

cout<<"\nDo you want to pay the bill to SUI-Gas ?";

cout<<"\n1. Yes \n2. No\n";

cin>>bill;

switch(bill)

{

case 1:

if(5000>account\_balance)

{

cout<<"\nNot enough balance";

cout<<"\nPress any key to continue";

getch();

}

else

{

system("cls");

mini\_gas="\n==>5000rs were paid to Sui-Gas\n";

account\_balance=account\_balance-5000;

cout<<"You have successfully paid the bill to SUI-Gas !\n";

cout<<"Press any key to continue";

getch();

}

break;

case 2:

system("cls");

cout<<"The bill was not paid!\n";

cout<<"Press any key to continue";

getch();

break;

default:

system("cls");

cout<<"Incorrect input\n";

cout<<"\nPress any key to continue";

getch();

break;

}

}

void electricity()

{

string address;

int bill;

system("cls");

cout<<"\nEnter your house address: ";

cin>>address;

getline(cin,address);

cout<<"\nYour bill is 35500rs";

cout<<"\nDo you want to pay the bill to K-electric ?";

cout<<"\n1. Yes \n2. No\n";

cin>>bill;

switch(bill)

{

case 1:

if(35500>account\_balance)

{

cout<<"\nNot enough balance";

cout<<"\nPress any key to continue";

getch();

}

else

{

system("cls");

mini\_electric="\n==>35000rs were paid to K-Electric\n";

account\_balance=account\_balance-35500;

cout<<"You have successfully paid the bill to K-electric ";

getch();

}

break;

case 2:

system("cls");

cout<<"The bill was not paid!\n";

cout<<"Press any key to continue";

getch();

break;

default:

system("cls");

cout<<"Incorrect input\n";

cout<<"\nPress any key to continue";

getch();

break;

}

}

void school(){

int school,STD\_ID,school\_fee;

system("cls");

Payment\_Title();

cout<<"\n1. City school\n";

cout<<"2. Allied school\n";

cout<<"3. Beacon house\n";

cout<<"4. The smart school\n";

cout<<"5. Back\n\n";

cout<<"Select Your school:";

cin>>school;

switch (school)

{

case 1:

system("cls");

Payment\_Title();

cout<<"\nEnter your student ID: ";

cin>>STD\_ID;

system("cls");

cout<<"\nYour due fee is: 2500rs";

cout<<"\nDo you want to pay the fee?";

cout<<"\n1. Yes\n2. No\n";

cin>>school\_fee;

switch (school\_fee)

{

case 1:

system("cls");

Payment\_Title();

if(2500>account\_balance)

{

cout<<"\nNot enough balance";

cout<<"\nPress any key to continue";

getch();

}

else

{

mini\_city="\n==>3000rs were paid to City School\n";

account\_balance=account\_balance-2500;

cout<<"You have successfully paid the fees !";

cout<<"\nPress any key to continue";

getch();

}

break;

case 2:

system("cls");

Payment\_Title();

cout<<"The fees was not paid!\n";

cout<<"Press any key to continue";

getch();

break;

default:

system("cls");

Payment\_Title();

cout<<"Incorrect input\n";

cout<<"\nPress any key to continue";

getch();

break;

}

break;

case 2:

system("cls");

Payment\_Title();

cout<<"\nEnter your student ID: ";

cin>>STD\_ID;

system("cls");

cout<<"\nYour due fee is: 5000rs";

cout<<"\nDo you want to pay the fee?";

cout<<"\n1. Yes\n2. No\n";

cin>>school\_fee;

switch (school\_fee)

{

case 1:

system("cls");

Payment\_Title();

if(5000>account\_balance)

{

cout<<"\nNot enough balance";

cout<<"\nPress any key to continue";

getch();

}

else

{

mini\_allied="\n==>5000rs were paid to Allied School\n";

account\_balance=account\_balance-5000;

cout<<"You have successfully paid the fees";

cout<<"\nPress any key to continue";

getch();

}

break;

case 2:

system("cls");

Payment\_Title();

cout<<"The fees was not paid!\n";

cout<<"Press any key to continue";

getch();

break;

default:

system("cls");

Payment\_Title();

cout<<"Incorrect input\n";

cout<<"\nPress any key to continue";

getch();

break;

}

break;

case 3:

system("cls");

Payment\_Title();

cout<<"\nEnter your student ID: ";

cin>>STD\_ID;

system("cls");

cout<<"\nYour due fees is: 10000rs";

cout<<"\nDo you want to pay the fees?";

cout<<"\n1. Yes\n2. No\n";

cin>>school\_fee;

switch (school\_fee)

{

case 1:

system("cls");

Payment\_Title();

if(10000>account\_balance)

{

cout<<"\nNot enough balance";

cout<<"\nPress any key to continue";

getch();

}

else

{

mini\_beacon="\n==>10000rs were paid to Beacon house School\n";

account\_balance=account\_balance-10000;

cout<<"You have successfully paid the fees";

cout<<"\nPress any key to continue";

getch();

}

break;

case 2:

system("cls");

Payment\_Title();

cout<<"The fees was not paid!\n";

cout<<"Press any key to continue";

getch();

break;

default:

system("cls");

Payment\_Title();

cout<<"Incorrect input\n";

cout<<"\nPress any key to continue";

getch();

}

break;

break;

case 4:

system("cls");

Payment\_Title();

cout<<"\nEnter your student ID: ";

cin>>STD\_ID;

system("cls");

cout<<"\nYour due fees is: 3000rs";

cout<<"\nDo you want to pay the fee?";

cout<<"\n1. Yes\n2. No\n";

cin>>school\_fee;

switch (school\_fee)

{

case 1:

system("cls");

Payment\_Title();

if(3000>account\_balance)

{

cout<<"\nNot enough balance";

cout<<"\nPress any key to continue";

getch();

}

else

{

mini\_smart="\n==>5000rs were paid to The smart School\n";

account\_balance=account\_balance-3000;

cout<<"You have successfully paid the fees";

cout<<"\nPress any key to continue";

getch();

}

break;

case 2:

system("cls");

Payment\_Title();

cout<<"The fees was not paid!\n";

cout<<"Press any key to continue";

getch();

break;

default:

system("cls");

Payment\_Title();

cout<<"Incorrect input\n";

cout<<"\nPress any key to continue";

getch();

break;

}

break;

case 5:

break;

default:

system("cls");

Payment\_Title();

cout<<"Incorrect input\n";

cout<<"\nPress any key to continue";

getch();

break;

}

}

void payment(){

Payment\_Title();

cout<<"1. School fees\n2. Electricity bill\n";

cout<<"3. Gas bill\n4. Water bill\n";

cout<<"5. back\n";

cout<<"Enter the action you want to perform: ";

int payment;

cin>>payment;

switch (payment)

{

case 1:

school();

break;

case 2:

electricity();

break;

case 3:

gas();

break;

case 4:

water();

break;

case 5:

break;

default:

system("cls");

cout<<"Incorrect input\n";

cout<<"\nPress any key to continue";

getch();

break;

}

}

void fastCash()

{

int fast\_cash;

FastCash\_Title();

cout<<"1. 500rs 2.1000rs\n";

cout<<"3. 3000rs 4.5000rs\n";

cout<<"5. 7000rs 6.15000rs\n";

cout<<"7. 25000rs 8.30000rs\n";

cout<<"9. Back\n";

cout<<"Select the action you want to perform: ";

cin>>fast\_cash;

cout<<"\n";

switch (fast\_cash)

{

case 1:

system("cls");

FastCash\_Title();

if(account\_balance<0)

{

cout<<"Not enough balance\n";

cout<<"Press any key to continue";

getch();

}

else

{

account\_balance=account\_balance-500;

mini\_cash1="\n==>500Rs were withdrawn\n";

cout<<"You have Successfully withdrawn 500rs\n\n";

cout<<"Press any key to continue";

getch();

}

break;

case 2:

system("cls");

FastCash\_Title();

if(account\_balance<0)

{

cout<<"Not enough balance\n";

cout<<"Press any key to continue";

getch();

}

else

{

account\_balance=account\_balance-1000;

mini\_cash2="\n==>1000Rs were with withdrawn\n";

cout<<"You have Successfully withdrawn 1000rs\n\n";

cout<<"Press any key to continue";

getch();

}

break;

case 3:

system("cls");

FastCash\_Title();

if(account\_balance<0)

{

cout<<"Not enough balance\n";

cout<<"Press any key to continue";

getch();

}

else

{

account\_balance=account\_balance-3000;

mini\_cash3="\n==>3000Rs were with withdrawn\n";

cout<<"You have Successfully withdrawn 3000rs\n\n";

cout<<"Press any key to continue";

getch();

}

break;

case 4:

system("cls");

FastCash\_Title();

if(account\_balance<0)

{

cout<<"Not enough balance\n";

cout<<"Press any key to continue";

getch();

}

else

{

account\_balance=account\_balance-5000;

mini\_cash4="\n==>5000Rs were with withdrawn\n";

cout<<"You have Successfully withdrawn 5000rs\n\n";

cout<<"Press any key to continue";

getch();

}

break;

case 5:

system("cls");

FastCash\_Title();

if(account\_balance<0)

{

cout<<"Not enough balance\n";

cout<<"Press any key to continue";

getch();

}

else

{

account\_balance=account\_balance-7000;

mini\_cash5="\n==>7000Rs were with withdrawn\n";

cout<<"You have Successfully withdrawn 7000rs\n\n";

cout<<"Press any key to continue";

getch();

}

break;

case 6:

system("cls");

FastCash\_Title();

if(account\_balance<0)

{

cout<<"Not enough balance\n";

cout<<"Press any key to continue";

getch();

}

else

{

account\_balance=account\_balance-15000;

mini\_cash6="\n==>15000Rs were with withdrawn\n";

cout<<"You have Successfully withdrawn 15000rs\n\n";

cout<<"Press any key to continue";

getch();

}

break;

case 7:

system("cls");

FastCash\_Title();

if(account\_balance<0)

{

cout<<"Not enough balance\n";

cout<<"Press any key to continue";

getch();

}

else

{

account\_balance=account\_balance-25000;

mini\_cash7="\n==>25000Rs were with withdrawn\n";

cout<<"You have Successfully withdrawn 25000rs\n\n";

cout<<"Press any key to continue";

getch();

}

break;

case 8:

system("cls");

FastCash\_Title();

if(account\_balance<0)

{

cout<<"Not enough balance\n";

cout<<"Press any key to continue";

getch();

}

else{

account\_balance=account\_balance-30000;

mini\_cash8="\n==>30000Rs were with withdrawn\n";

cout<<"You have Successfully withdrawn 30000rs\n\n";

cout<<"Press any key to continue";

getch();

}

break;

case 9:

break;

default:

system("cls");

cout<<"Incorrect Input\n\n";

cout<<"\nPress any key to continue";

getch();

break;

}

}

void withdraw()

{

Withdraw\_Title();

cout<<"Enter the amount of money you want to withdraw:";

cin>>cash\_withdraw;

if(cash\_withdraw>account\_balance)

{

cout<<"Not enough balance";

cout<<"\nPress any key to continue";

getch();

}

else

{

system("cls");

Withdraw\_Title();

cout<<"\nYou have successfully withdrawn "<<cash\_withdraw<<"rs\n";

cout<<"\nPress any key to continue";

account\_balance=account\_balance-cash\_withdraw;

getch();

}

}

void funds\_transfer()

{

Funds\_Transfer();

cout<<"Enter the account No. you want to transfer funds to:";

cin>>account\_number;

cout<<"Enter the amount of money you want to transfer: ";

cin>>funds;

if(funds>account\_balance)

{

system("cls");

cout<<"\nNot enough balance";

cout<<"\nPress any key to continue";

getch();

}

else

{

system("cls");

Funds\_Transfer();

account\_balance=account\_balance-funds;

cout<<"\nYou have successfully transfered "<<funds<<"rs to the account number "<<account\_number<<"\n";

cout<<"\nPress any key to continue";

getch();

}

}

void mini\_statement()

{

cout<<mini<<mini\_cash1<<mini\_cash2<<mini\_cash3<<mini\_cash4<<mini\_cash5<<mini\_cash6<<mini\_cash7<<mini\_cash8;

if(cash\_withdraw>0)

{

cout<<"\n==>"<<cash\_withdraw<<"Rs were with wihtdrawn\n";

}

if(funds>0)

{

cout<<"\n==>"<<funds<<"Rs were transfered to account No. "<<account\_number<<"\n";

}

cout<<mini\_city<<mini\_allied<<mini\_beacon<<mini\_smart<<mini\_electric<<mini\_gas<<mini\_water;

cout<<"\nPress any key to continue";

getch();

}

***ALGORITHM:***

1.) In order to continue, the user must provide their User ID and Password on the program's login screen. If either piece of information is entered incorrectly, a message reading **"Incorrect input"** is printed. **"practice2.txt"** is the file where this data is kept.

2.) The program will direct you to a new screen where you can carry out various operations once your User ID and password have been verified.

3.) The data in this menu-driven program is controlled by **Switch Case** on the New screen (Main Menu).

4.) The user will be able to withdraw a set amount of money in the first case.

5.) The user may withdraw the desired quantity of money in the second case.

6.) The user may view their Account balance in the third case. Every time a user conducts any activity involving money, the account balance is altered. Utilizing the **"If else"** condition, this is done.

7.) Fourth case allows the user to check their mini-statement which is done through **“if”** function. Mini-statement is updated when ever user performs a action involving cash.

8The user is first requested to enter his existing password in the fifth case, which allows password changes. A message reading "Wrong password" will be printed if the current password is incorrect. The user will be prompted to enter his new password if the password is accurate. Using **"go to label,"** the user is returned to the login screen after entering a new password. A file called **"practice2.txt"** contains all of the User IDs and passwords.

8.) Sixth case allows the user to transfer funds to another account.

9.) The user may pay their utility, gas, water, and school bills in the seventh case. **"Switch Case"** is the tool used to create all of these features. Since there are numerous schools to choose from, switch case is also used in the school function().

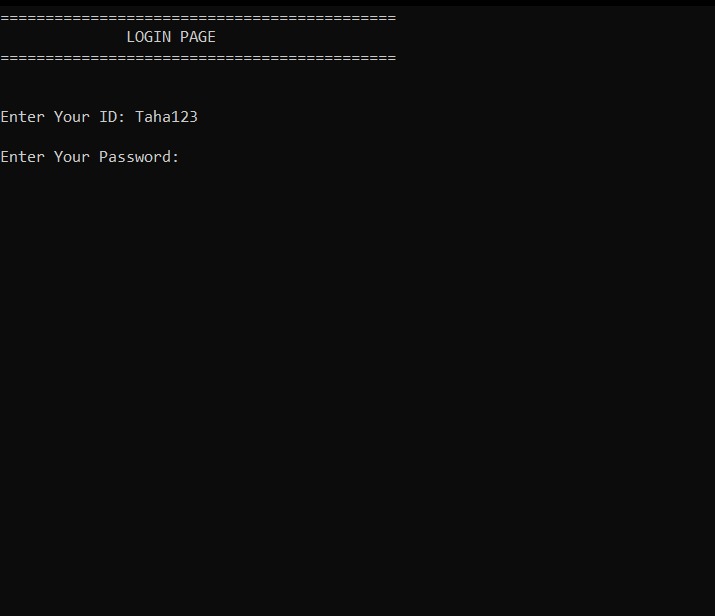
10.) The user may log out in the eighth and final case so that another user may sign in. To accomplish this, **"go to label"** is used.

11.) Every time a user inputs an inaccurate input, the text **"Incorrect Input"** is generated, and the user is then returned to the main menu panel.

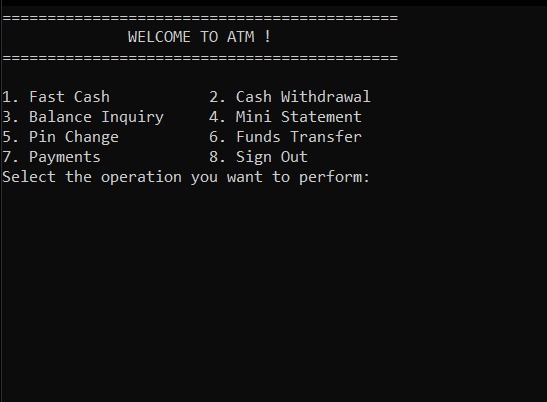
12) Using the **"go to label"** command, the user is repeatedly taken back to the program's main screen after completing each action.

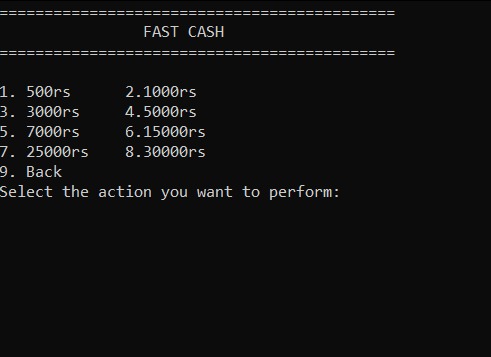
***RESULTS:***

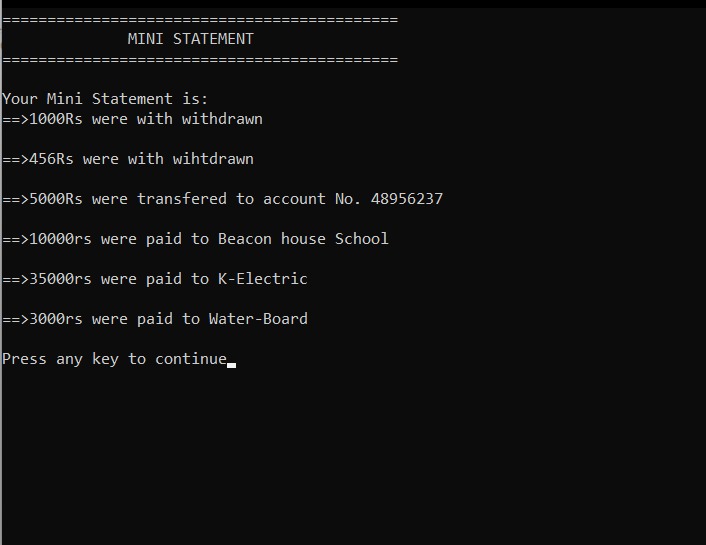
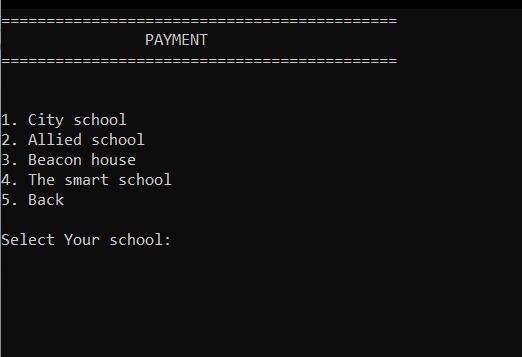
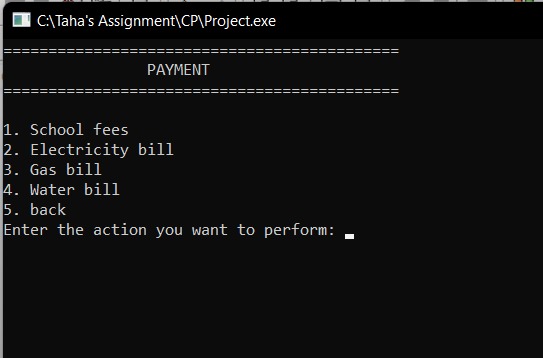
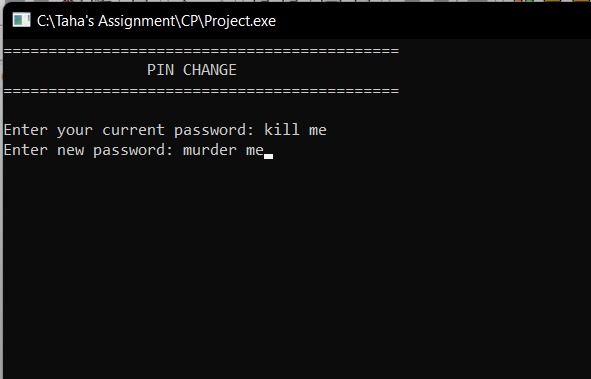
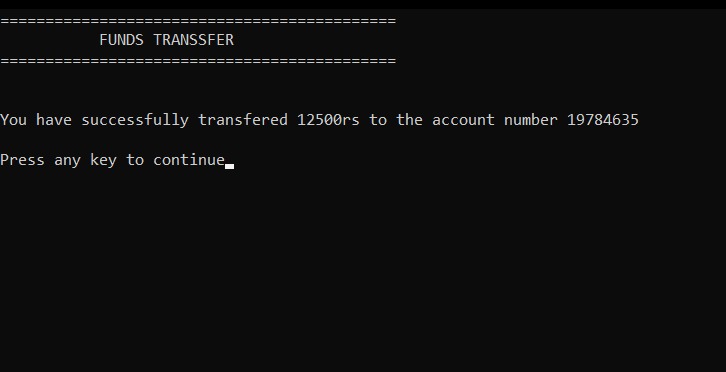
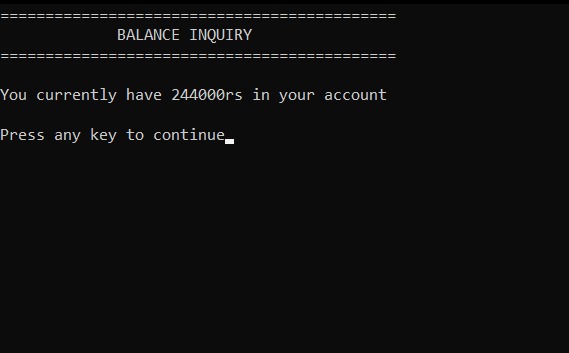
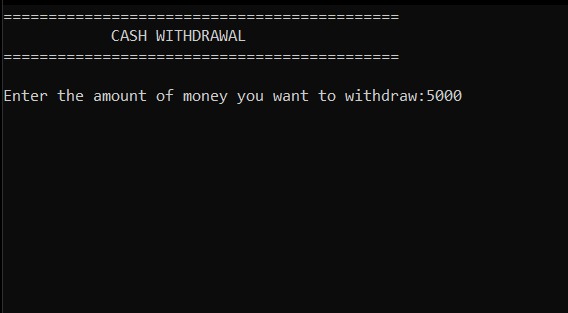
***LOGIN PAGE:***

******

***MAIN MENU:***

******

******

******

***CONCLUSION*:**

It was a fantastic experience in which we learned a lot of new things. It felt difficult at first, but as the project progressed, it became easier, and we were very happy each time a function we created was executed without error; it gave us a sense of accomplishment. The switch statement was the easiest part, while filing was the most difficult because we hadn't learned it in university yet. I had no idea I would have to learn filing in order to finish the project, and it was quite interesting. It has a lot of potential for new applications. If I had to redo the project, I might do things differently. Someone told me I could efficiently use structures in this program but I choose not to since I haven’t learned it yet.