Bank Management System

Submitted as Mini Project for Special Topics: Programming with C++ Semester III

**BACHELOR OF TECHNOLOGY**

in

Computer Science and Engineering

**Monish Prakasan**

PES1201700232

**Pushpender Singh**

PES1201700243

**Pranjal Srivastava**

PES1201700208

Under Guidance of

Prof. Jamuna S Murthy

BE (CSE), M.Tech (SSE)

Assistant Professor

**Aug 2018- Dec 2018**

**Department of Computer Science and Engineering**

**PES UNIVERSITY**

Outer Ring Rd, Banashankari 3rd Stage, Banashankari, Bengaluru, Karnataka 560085

[www.pes.edu](http://www.pes.edu)

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| SLNO | TOPIC | PAGE |
| 1) | OBJECTIVE | 1 |
| 2) | SYSTEM CONFIGURATION | 2 |
| 3) | INTRODUCTION | 3 |
| 4) | THEORY | 4 |
| 5) | FUNCTION DESCRIPTION | 7 |
| 6) | SOURCE | 8 |
| 7) | SCREENSHOTS | 15 |

**ACKNOWLEDGEMENT**

We thank our Computer Science teacher Ms. Jamuna S Murthy for her guidance and support. We would also like to thank our parents and other family members for encouraging us during the course of this project. Finally we would like to thank PES University for giving us the opportunity to undertake this project.

SYSTEM CONFIGURATION

|  |  |
| --- | --- |
| FIELD | CONFIGURATION |
| OPERATING SYSTEM  CPU  RAM  SYSTEM MAKE TYPE  GCC VERSION | Windows 10 Pro  Intel Core i5  2450M  2 Cores @2.50GHz  (4 Logical Processors)  4 GB DDR3  Intel 3000HD  Radeon 7670m HD  HP Pavilion g6 2006-tx  X64-based PC  MinGW  Gcc 6.2.0 |

SOURCE

//NOTE-Password :compbank

#include<conio.h>

#include<iostream>

#include<fstream>

#include<stdio.h>

#include<string.h>

using namespace std;

intaccbuf;

string pass ="";

void password(){

    cout<<"Enter the PASSWORD to verify that you are an ADMINISTRATOR :\n";

    charch;

    ch = \_getch();

   while(ch != 13){//character 13 is enter

      pass.push\_back(ch);

      cout<< '\*';

      ch = \_getch();

   }

}

class Bank{

    long accno;

    float amount;   char type;

public:

    char name[30];

    void input(){

        char sel;

        repeat:

        cout<<"Enter the Account Number : ";   cin>>accno; cout<<"Accno : "<<accno<<endl;

        cout<<"Enter Name of the Account Holder : ";  cin.ignore(); cin.getline(name,30);

        cout<<"Enter Type of the Account (Savings(s)/Checking(c)) : ";   cin>>type;

        cout<<"Enter the Initial Amount :\n>=500 for Savings &>=1000 for Checking : ";  cin>>amount;

        if(type=='s'||type=='S'){

            if(amount<=500){

                cout<<"Invalid Amount for Savings Account!\nTRY AGAIN!!"<<endl;

                goto repeat;

                }

        }

        else if(type=='c'||type=='C'){

            if(amount<=1000){

                cout<<"Invalid Amount for Checking Account!\nTRY AGAIN!!"<<endl;

                goto repeat;

                }

        }

    }

    long getaccno(){

        returnaccno;

    }

    float getamt(){

        return amount;

    }

    float dep\_amount(float a){

        amount+=a;

    }

    float wd\_amount(float a){

        amount-=a;

    }

};

Bank x,y;

void new\_account(){

    pass ="";

    password();

    if(pass == "compbank"){

      cout<< "\nAccess granted \n";

      cout<<"\*\*\*\*\*\*\*\*\*\*NEW ACCOUNT\*\*\*\*\*\*\*\*\*\*\n\n";

      repeat\_entry:

        ofstream fout("Bank.dat",ios::app);

        cout<<"Enter details for the NEW ACCOUNT : "<<endl;

        x.input();

        ifstream fin("Bank.dat");

        while(!fin.eof()){

            fin.read((char\*)&y,sizeof(Bank));

            long accbuf1=y.getaccno();

            long accbuf2=x.getaccno();

            if(accbuf1==accbuf2){

                cout<<"CONFLICTING ACCOUNT NUMBERS! TRY AGAIN!"<<endl;

                gotorepeat\_entry;

            }

        fin.close();

        fout.write((char\*)&x,sizeof(Bank));

        fout.close();

        }

   }

   else{

      cout<< "\nAccess Denied\n";

   }

}

voiddeposit\_amount(){

    pass ="";

    password();

    if(pass == "compbank"){

        cout<< "\nAccess granted \n";

        cout<<"\*\*\*\*\*\*\*\*\*\*DEPOSIT AMOUNT\*\*\*\*\*\*\*\*\*\*\n\n";

        floatdep\_amt; long accno,buffer;

        ofstreamfout("temp.dat",ios::app);

        cout<<"Enter the Account Number to deposit the amount to : ";   cin>>accno;

        ifstream fin("Bank.dat",ios::in);

        while(!fin.eof()){

            fin.read((char\*)&x,sizeof(Bank));

            longaccbuf=x.getaccno();

            if(accno==accbuf){

                if(buffer==accbuf)  goto over;

                cout<<"Enter the Amount to deposit : "; cin>>dep\_amt;

                x.dep\_amount(dep\_amt);

                cout<<"DEPOSITED "<<dep\_amt<<" to "<<[x.name](http://x.name/)<<"'s Account"<<endl;

                fout.write((char\*)&x,sizeof(Bank));

                over:

                buffer=accbuf;

            }

            else

                fout.write((char\*)&x,sizeof(Bank));

        }

        cout<<"\n\n";

        fout.close();

        fin.close();

        remove("Bank.dat");

        rename("Temp.dat","Bank.dat");

        cout<<"\n\n";

    }

    else

        cout<<"\nACCESS DENIED"<<endl;

}

void withdraw\_amount(){

    pass ="";

    password();

    if(pass == "compbank"){

        cout<< "\nAccess granted \n";

        cout<<"\*\*\*\*\*\*\*\*\*\*WITHDRAW AMOUNT\*\*\*\*\*\*\*\*\*\*\n\n";

        float wd\_amt; long accno,buffer;

        ofstream fout("temp.dat",ios::app);

        cout<<"Enter the Account Number to withdraw the amount from : ";    cin>>accno;

        ifstream fin("Bank.dat",ios::in);

        while(!fin.eof()){

            fin.read((char\*)&x,sizeof(Bank));

            long accbuf=x.getaccno();

            if(accno==accbuf){

                if(buffer==accbuf)  goto over2;

                cout<<"Enter the Amount to withdraw : "; cin>>wd\_amt;

                x.wd\_amount(wd\_amt);

                cout<<"WITHDRAWED "<<wd\_amt<<" from "<<[x.name](http://x.name/)<<"'s Account"<<endl;

                fout.write((char\*)&x,sizeof(Bank));

            over2:

                buffer=accno;

            }

            else

                fout.write((char\*)&x,sizeof(Bank));

        }

        cout<<"\n\n";

        fout.close();

        fin.close();

        remove("Bank.dat");

        rename("Temp.dat","Bank.dat");

        cout<<"\n\n";

    }

    else

        cout<<"\nACCESS DENIED"<<endl;

}

void balance(){

        cout<<"\n\n"<<"\*\*\*\*\*\*\*\*\*\*BALANCE ENQURIY\*\*\*\*\*\*\*\*\*\*"<<"\n\n";

        longaccno; char namebuf[30];

        cout<<"Enter the Account Number to check its Balance : ";   cin>>accno;

        ifstream fin("Bank.dat",ios::in);

        while(!fin.eof()){

            fin.read((char\*)&x,sizeof(Bank));

            longaccbuf=x.getaccno();

            if(accno==accbuf){

                float bal=x.getamt();

                cout<<"Account Balance is : "<<bal<<endl;

                goto end\_of\_balance;

            }

        }

        end\_of\_balance:

        fin.close();

}

Void account\_holders() {

    cout<<"\n\n\*\*\*\*\*\*\*\*\*\*ACCOUNT HOLDERS\*\*\*\*\*\*\*\*\*\*\n\n";

    ifstream fin("Bank.dat",ios::in); int counter=1; long accno,buffer;

    while(!fin.eof()){

        fin.read((char\*)&x,sizeof(Bank));

        accno=x.getaccno();

        if(buffer==accno)

            goto endofloop;

        cout<<counter<<". "<<[x.name](http://x.name/)<<endl;

        counter++;

            endofloop:

                buffer=accno;

                ;

    }

    cout<<"\n\n";

    fin.close();

}

voidmodify\_account() {

    pass ="";

    password();

    if(pass == "compbank"){

        cout<< "\nAccess granted \n";

        cout<<"\*\*\*\*\*\*\*\*\*\*ACCOUNT MODIFIER\*\*\*\*\*\*\*\*\*\*\n\n";

        longaccno,dummy=0; char namebuf[30];

        ofstreamfout("temp.dat",ios::app);

        cout<<"Enter the Account Number to MODIFY its contents : ";   cin>>accno;

        ifstream fin("Bank.dat",ios::in);

        while(!fin.eof()){

            fin.read((char\*)&x,sizeof(Bank));

            if(x.getaccno()==dummy)

                gotoend\_of\_modify;

            longaccbuf=x.getaccno();

            if(accno==accbuf){

                cout<<"Enter the new details : "<<endl;

                x.input();

                dummy=x.getaccno();

                fout.write((char\*)&x,sizeof(Bank));

            }

            else{

                fout.write((char\*)&x,sizeof(Bank));

                dummy=x.getaccno();

            }

        }

        end\_of\_modify:

        cout<<"\n\n";

        fout.close();

        fin.close();

        remove("Bank.dat");

        rename("Temp.dat","Bank.dat");

        cout<<"\n\n";

    }

    else

        cout<<"\nACCESS DENIED"<<endl;

}

voidclose\_account() {

    pass ="";

    password();

    if(pass == "compbank"){

        cout<< "\nAccess granted \n";

        cout<<"\*\*\*\*\*\*\*\*\*\*ACCOUNT CLOSER\*\*\*\*\*\*\*\*\*\*\n\n";

        longaccno,buffer;   int counter=0;

    ofstreamfout("temp.dat",ios::out);

    cout<<"Enter the Account Number to Close The Account : ";    cin>>accno;

    ifstream fin("Bank.dat",ios::in);

    while(!fin.eof()){

        fin.read((char\*)&x,sizeof(Bank));

        longaccbuf=x.getaccno();

        if(accno==accbuf){

            if(buffer==accno)   gotoend\_of\_loop;

            charch;

            cout<<"Are you sure ? (y/n) "; cin>>ch;

            if(ch=='y'||ch=='Y'){

                cout<<"CLOSED THE ACCOUNT"<<endl;

                counter++;

            }

            else{

                fout.write((char\*)&x,sizeof(Bank));

                counter++;

            }

            end\_of\_loop:

                buffer=accno;

                    ;

        }

        else

            fout.write((char\*)&x,sizeof(Bank));

    }

    cout<<"\n\n";

    fin.close();

    fout.close();

    remove("Bank.dat");

    rename("Temp.dat","Bank.dat");

    cout<<"\n\n";

    }

    else

        cout<<"\nACCESS DENIED"<<endl;

}

int main(){ char art1[]=" $$$$$$\\     $$$$$\\  $$$$$$\\  $$\\   $$\\       \n$$  \_\_$$\\    \\\_\_$$ |$$  \_\_$$\\ $$ |  $$ |      \n$$ /  $$ |      $$ |$$ /  $$ |\\$$\\ $$  |      \n$$$$$$$$ |      $$ |$$$$$$$$ | \\$$$$  /       \n$$  \_\_$$ |$$\\   $$ |$$  \_\_$$ | $$  $$<        \n$$ |  $$ |$$ |  $$ |$$ |  $$ |$$  /\\$$\\       \n$$ |  $$ |\\$$$$$$  |$$ |  $$ |$$ /  $$ |      \n\\\_\_|  \\\_\_| \\\_\_\_\_\_\_/ \\\_\_|  \\\_\_|\\\_\_|  \\\_\_|      \n                                              \n                                              \n                                              \n\n";

                    char art2[]="$$\\   $$\\  $$$$$$\\ $$$$$$$$\\ $$$$$$\\  $$$$$$\\  $$\\   $$\\  $$$$$$\\  $$\\       \n$$$\\  $$ |$$  \_\_$$\\\\\_\_$$  \_\_|\\\_$$  \_|$$  \_\_$$\\ $$$\\  $$ |$$  \_\_$$\\ $$ |      \n$$$$\\ $$ |$$ /  $$ |  $$ |     $$ |  $$ /  $$ |$$$$\\ $$ |$$ /  $$ |$$ |      \n$$ $$\\$$ |$$$$$$$$ |  $$ |     $$ |  $$ |  $$ |$$ $$\\$$ |$$$$$$$$ |$$ |      \n$$ \\$$$$ |$$  \_\_$$ |  $$ |     $$ |  $$ |  $$ |$$ \\$$$$ |$$  \_\_$$ |$$ |      \n$$ |\\$$$ |$$ |  $$ |  $$ |     $$ |  $$ |  $$ |$$ |\\$$$ |$$ |  $$ |$$ |      \n$$ | \\$$ |$$ |  $$ |  $$ |   $$$$$$\\  $$$$$$  |$$ | \\$$ |$$ |  $$ |$$$$$$$$\\ \n\\\_\_|  \\\_\_|\\\_\_|  \\\_\_|  \\\_\_|   \\\_\_\_\_\_\_| \\\_\_\_\_\_\_/ \\\_\_|  \\\_\_|\\\_\_|  \\\_\_|\\\_\_\_\_\_\_\_\_|\n                                                                             \n                                                                             \n                                                                             \n\n";

                    char art3[]="$$$$$$$\\   $$$$$$\\  $$\\   $$\\ $$\\   $$\\ \n$$  \_\_$$\\ $$  \_\_$$\\ $$$\\  $$ |$$ | $$  |\n$$ |  $$ |$$ /  $$ |$$$$\\ $$ |$$ |$$  / \n$$$$$$$\\ |$$$$$$$$ |$$ $$\\$$ |$$$$$  /  \n$$  \_\_$$\\ $$  \_\_$$ |$$ \\$$$$ |$$  $$<   \n$$ |  $$ |$$ |  $$ |$$ |\\$$$ |$$ |\\$$\\  \n$$$$$$$  |$$ |  $$ |$$ | \\$$ |$$ | \\$$\\ \n\\\_\_\_\_\_\_\_/ \\\_\_|  \\\_\_|\\\_\_|  \\\_\_|\\\_\_|  \\\_\_|\n                                        \n                                        \n                                        \n\n";

    cout<<art1<<art2<<art3;

    menu:

    cout<<"\*\*\*\*\*\*\*\*\*\*BANKING SYSTEM\*\*\*\*\*\*\*\*\*\*\n\n";

    cout<<"-----MAIN MENU-----\n";

    cout<<"1. New Account\n";

    cout<<"2. Deposit Amount\n";

    cout<<"3. Withdraw Amount\n";

    cout<<"4. Balance Enquiry\n";

    cout<<"5. All Account Holders List\n";

    cout<<"6. Close an Account\n";

    cout<<"7. Modify Account\n";

    cout<<"--Any Other Key to Exit--\n\n\n";

    int n;  cin>>n;

    switch(n){

        case 1: new\_account();

        goto menu;

        case 2: deposit\_amount();

        goto menu;

        case 3: withdraw\_amount();

        goto menu;

        case 4: balance();

        goto menu;

        case 5:account\_holders();

        goto menu;

        case 6:close\_account();

        goto menu;

        case 7:modify\_account();

        goto menu;

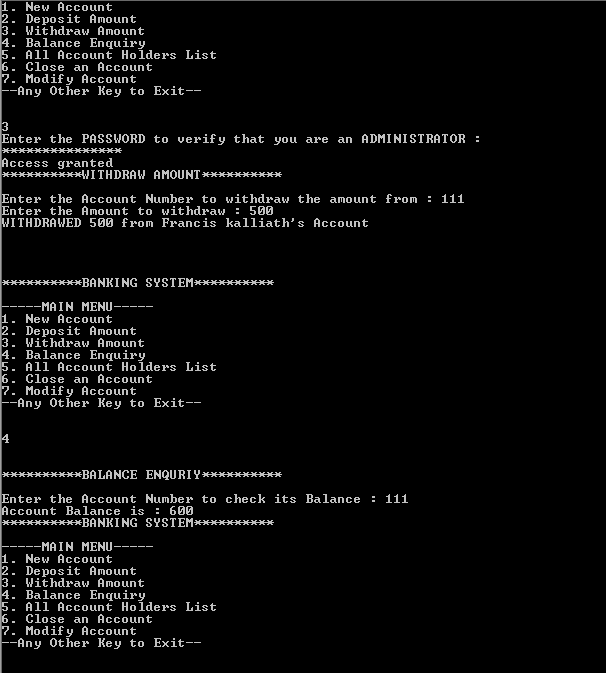
    }

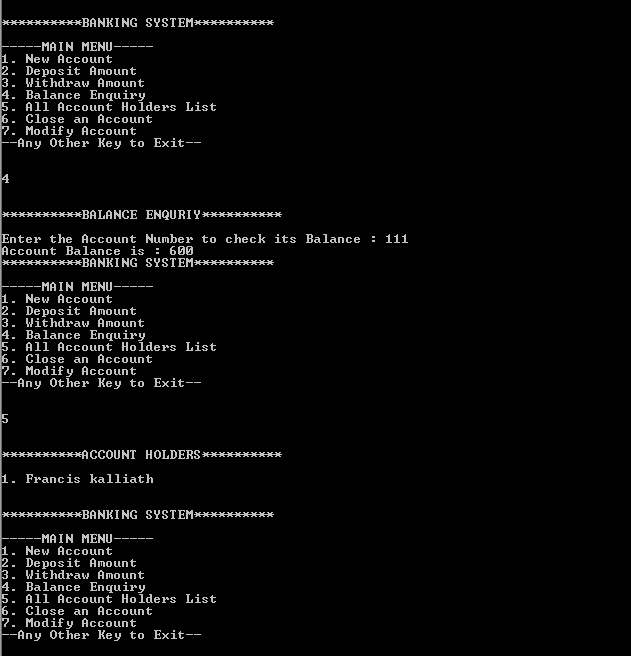
    return 0;

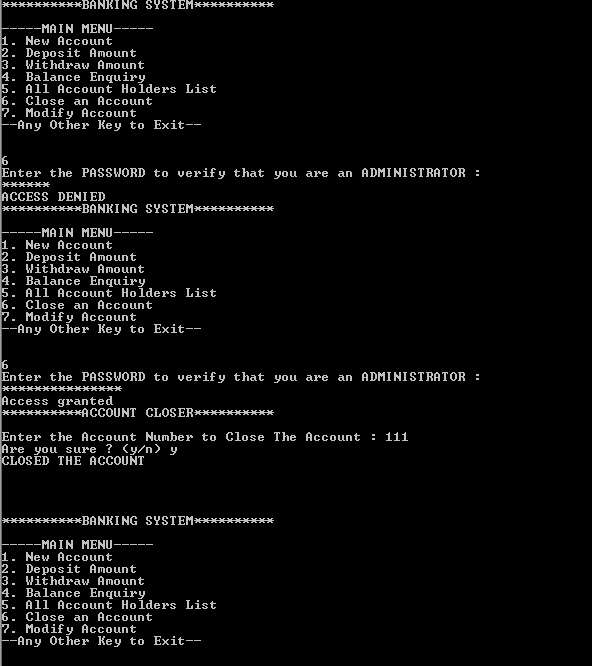
}

SCREENSHOTS











FUNCTION DESCRIPTION

New account - creates new account, collects details from the user and writes it to the external file.

Deposit money - this function is used to add/allocate money to a user's bank account

Withdraw money - this function is used to deduct/withdraw money from a person's account.

Modify account - this function is user to update an account holder's details.

Close account - this function is used to close a person's account and remove it completely from the external database.

INTRODUCTION

C++ is an object oriented programming (OOP) language, developed by Bjarne Stroustrup, and is an extension of C language. It is therefore possible to code C++ in a "C style" or "object-oriented style." In certain scenarios, it can be coded in either way and is thus an effective example of a hybrid language. The main highlight of C++ is a collection of pre-defined classes, which are data types that can be instantiated multiple times. The language also facilitates declaration of user defined classes. Classes can further accommodate member functions to implement specific functionality. Multiple objects of a particular class can be defined to implement the functions within the class. Objects can be defined as instances created at run time. These classes can also be inherited by other new classes which take in the public and protected functionalities by default. The banking system we created is called “Ajax National Bank”.

This banking system program uses very basic c++ functions to mimic the working of a conventional banking system.

Most of the header files used are the commonly used ones and the program is easy to comprehend and understand. This is a system which is highly user friendly and is to be used by the customers who visit this bank.

THEORY

In the program, to perform certain functions like creating a bank account, or withdrawing/depositing money, etc, which require administrator privileges in the real world, can be invoked only upon entering the password.

The password for this program is - compbank.

The given program is nothing but a mild interpretation of the heavily secure and complex banking systems of today.

Also one more function - account holder's list - this function can be used to display the account holders of the bank. No password is required here.

The list of header files used in our project is listed below.

1. **FSTREAM.H** – for file handling, cin and cout
2. **CONIO.H** – for clrscr() and getch() functions
3. **STDIO.H** – for standard I/O operations
4. **STRING.H** – for string handling
5. **IOSTREAM.H –** For Standard I/O Operations

This program uses the concept of data file handling. In data **file handlin**g data can be permanently stored in the computer. Using this we can store our data in Secondary memory (Hard disk). It has been made into a menu based program for the benefit of the user.

OBJECTIVE

TO CREATE A BANKING SYSTEM USING C++

MENU FLOWCHART

