```cpp
//BANKING MANAGEMENT SYSTEM WITH FILE HANDLING
//Programmed by: BSCpE 2-1 Group 1
//Esteban, Charlene Mae DG.
//Gomez, Enrico Camilo P. Jr.
//Guevarra, Justin Earl L.
//Morita, Ami V.
//Submitted to: Engr. Julius S. Cansino

#include <iostream>
#include <fstream>
#include <cctype>
#include <iomanip>
#include <Windows.h>
#include <string>
#include <time.h>
#include <cstdlib>
#include <conio.h>
using namespace std;

int usernum;
struct account
{

    int accNo;
    int accPin;
    char accName[50];
    float accBalance;
    char accType;
    void createAccount();
    void show() const;
    void modify(int);
    void deposit(int);
    void withdraw(int);
    void accDetails(int) const;
    int retaccPin() const;
    int retaccNo() const;
    int retaccBalance() const;
    char retaccType() const;
}ac;

//Functions needed in file handling******************
void writeAccount();    //function to write record in binary file
void displayAccount(int);   //function to display account details given by user
void modifyAccount(int,int);    //function to modify record of file
void deleteAccount(int);    //function to delete record of file
void dispAll();     //function to display all account details
void depwith(int, int); // function to desposit/withdraw amount for given account
void welcome();
void logoandtime();
int login();
void matchAccount(int);

//FUNCTIONS
void pos(int x, int y){
    COORD coord;
    coord.X = x;
    coord.Y = y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE),coord);
}

void col(int k){
    HANDLE hConsole = GetStdHandle(STD_OUTPUT_HANDLE);
    SetConsoleTextAttribute(hConsole, k);
}

const std::string currentDateTime() {
    time_t     now = time(0);
    struct tm  tstruct;
    char       buf[80];
    tstruct = *localtime(&now);
    strftime(buf, sizeof(buf), "%X   %B %d, %Y", &tstruct);
    return buf;
}

void logoandtime(){
    col(7);pos(1,1);cout <<currentDateTime();
    col(205);pos(90,1);cout<<"   ";
    pos(90,2);cout<<" ";
    pos(90,3);cout<<"   ";
    pos(92,2);cout<<" ";
    pos(94,1);cout<<" ";
    pos(94,2);cout<<" ";
    pos(94,3);cout<<"   ";
    col(240);pos(90,4); cout<< "B A N K";
    col(15);
}

void account::createAccount()
{
    logoandtime();
    col(112);cout<<"\n\tAccount No.: ";
    accNo=rand()%1000 + 1500; //for random making of account numbers

    bool found=false;

    fstream File;
    File.open("accdata.dat",ios::binary|ios::in|ios::out);
    while(!File.eof() && found==false)
    {
        File.read(reinterpret_cast<char *> (&ac), sizeof(account));
        if(ac.retaccNo()==accNo)
        {
                accNo=rand()%1000 + 1500;
          }
    }
    File.close();

    cout<<accNo;
        col(15);

```

```
int tempPin1, tempPin2, loop1;
cout << "\n\tEnter your PIN: ";
cin >> tempPin1;
do{
    loop1=1;
    cout <<"\tRetype PIN: ";
    cin >>tempPin2;
    if(tempPin1==tempPin2)
        accPin=tempPin1;
    else loop1=0;
}while(loop1==0);


int x=0,loop=0;
char tempaccName[50];
do
{
    cout<<"\n\n\tFull Name: ";
    cin.ignore();
    cin.getline(tempaccName,50);
    x=0;
    for(int i=0 ; i < strlen(tempaccName); i++){
        if(!isalpha(tempaccName[i]) ){
            loop = 1;
            x++;
            if(isspace(tempaccName[i])){
                x--;
            }
            if(ispunct(tempaccName[i])){
                x--;
            }
        }
    }
    if(x==0){
        strcpy(accName, tempaccName);
        loop = 0;
    }
    else {
        col(12);cout <<"\tInvalid Name!";col(15);
    }
}while(loop == 1);

x=1;
do{
    cout<<"\n\tEnter Type of Account (C/S) Current or Savings: ";
    accType=_getch();
    accType=toupper(accType);
    cout <<accType<<endl;
    if((accType == 'S')||(accType=='C'))
        x=0;
    else {
        col(12);cout << "\tPlease Enter a Valid Type of Account!";col(15);}
}while(x==1);
loop=0;
do{
```

```cpp
    cout<<"\n\tEnter the Initial Deposit Amount: Php ";
    cin>>accBalance;
    if ((accType=='S')&&(accBalance<500))
    {
            col(12);cout<<"\n\tRequired Amount for Savings Account is Php500.00";col(15);
            loop=1;
    }
    else if((accType=='C')&&(accBalance<1000))
    {
        col(12);cout<<"\n\tRequired Amount for Current Account is Php1000.00";col(15);
        loop=1;
    }
    else loop=0;
    }while(loop==1);

    col(121);cout<<"\n\n\t                    ";
    cout<<"\n\t  ACCOUNT CREATED!  ";
    cout<<"\n\t                    ";col(15);
}

void account::show() const
{
    system("cls");
    logoandtime();
    col(112);cout<<"\n\tAccount No.: "<<accNo;
    col(15);
    cout<<"\n\n\tAccount Holder Name: ";
    cout<<accName;
    cout<<"\n\tType of Account: "<<accType;
    cout<<"\n\tRemaining Balance: Php "<<fixed<<setprecision(2)<<accBalance<<endl<<endl;
    cout<<"\t";system("pause");
}


void account::modify(int x)
{
    system("cls");
    logoandtime();

    if(x==1){

    col(112); cout<<"\n\n\tModify Account Holder Name: "; col(15);

    int x=0,loop=0;
    char tempaccName[50];
    do
    {
        cin.ignore();
        cin.getline(tempaccName,50);
        x=0;
            for(int i=0 ; i < strlen(tempaccName); i++) {
                if(!isalpha(tempaccName[i]) )   {
                    loop = 1;
                    x++;
                    if(isspace(tempaccName[i])){
```

```
                    x--;
                    }
                    if(ispunct(tempaccName[i])){
                    x--;
                    }
                }
            }

    if(x==0){
        strcpy(accName, tempaccName);
        loop = 0;
    }
    else {
        col(12);cout <<"\tEnter a Valid Name: ";
    }


    }while(loop == 1);
    col(112); cout<<"\n\tModify Account PIN: "; col(15);
    cin >> accPin;}
    else if(x==2){
    col(112); cout<<"\n\n\tModify Account PIN: "; col(15);
    cin >> accPin;}

}

void account::deposit(int x)
{
    accBalance+=x;
}

void account::withdraw(int x)
{
    accBalance-=x;
}

void account::accDetails(int x) const
{
    pos(10,x);cout<<accNo;
    pos(25,x);cout<<accName;
    pos(51,x);cout<<accType;
    pos(59,x);cout<<"Php "<<fixed<<setprecision(2)<<accBalance;
}

int account::retaccNo() const
{
    return accNo;
}

int account::retaccBalance() const
{
    return accBalance;
}

char account::retaccType() const
```

```cpp
{
    return accType;
}

int account::retaccPin() const
{
    return accPin;
}

void writeAccount()
{


    ofstream outFile;
    outFile.open("accdata.dat",ios::binary|ios::app);
    ac.createAccount();
    outFile.write(reinterpret_cast<char *> (&ac), sizeof(account));
    outFile.close();
}

//Functions to read specific record from file

void displayAccount(int n)
{
    logoandtime();

    bool flag=false;
    ifstream inFile;
    inFile.open("accdata.dat",ios::binary);
    if(!inFile)
    {
        col(12);pos(2,12);cout<<"\n\tNo records found!";col(15);
        return;
    }
    system("cls");
    cout<<"\n\tBALANCE DETAILS\n";

        while(inFile.read(reinterpret_cast<char *> (&ac), sizeof(account)))
    {
        if(ac.retaccNo()==n)
        {
            ac.show();
            flag=true;
        }
    }
    inFile.close();
    if(flag==false)
    {
        col(124);cout<<"\n\n\t                           ";
                 cout<<"\n\t  ACCOUNT DOES NOT EXIST!  ";
                 cout<<"\n\t                           ";col(15);
    }
}

//Function to modify record of file
```

```
void modifyAccount(int n,int m)
{
    bool found=false;

    fstream File;
    File.open("accdata.dat",ios::binary|ios::in|ios::out);
    if(!File)
    {
        col(12);pos(2,12);cout<<"\n\tNo records found!";col(15);
        return;
    }
    while(!File.eof() && found==false)
    {
        File.read(reinterpret_cast<char *> (&ac), sizeof(account));
        if(ac.retaccNo()==n)
        {
            ac.show();
            ac.modify(m);
            int pos=(-1)*static_cast<int>(sizeof(account));
            File.seekp(pos,ios::cur);
            File.write(reinterpret_cast<char *> (&ac), sizeof(account));
            col(121);cout<<"\n\n\t                   ";
            cout<<"\n\t  RECORD UPDATED!  ";
            cout<<"\n\t                   ";col(15);
            found=true;
          }
    }
    File.close();
    if(found==false){
        col(124);cout<<"\n\n\t                           ";
                 cout<<"\n\t  ACCOUNT DOES NOT EXIST!  ";
                 cout<<"\n\t                           ";col(15);}
}

//Function to delete record of file
void deleteAccount(int n)
{

    ///////////////
    bool found=false;
    fstream File;
    File.open("accdata.dat",ios::binary|ios::in|ios::out);
    while(!File.eof() && found==false)
    {
        File.read(reinterpret_cast<char *> (&ac), sizeof(account));
        if(ac.retaccNo()==n)
        {
        ac.show();
        found=true;}
    }
    File.close();
    if(found==false)
    {
        col(124);cout<<"\n\n\t                           ";
                 cout<<"\n\t  ACCOUNT DOES NOT EXIST!  ";
                 cout<<"\n\t                           ";col(15);
            }
    else
    {
        col(121);cout<<"\n\n\t                   ";
                 cout<<"\n\t  ACCOUNT CLOSED!  ";
                 cout<<"\n\t                   ";col(15);
    }
        //////////////
    ifstream inFile;
    ofstream outFile;
    inFile.open("accdata.dat",ios::binary);
    if(!inFile)
    {
        cout<<"File could not be open !! Press any Key...";
        return;
    }
    outFile.open("Temp.dat",ios::binary);
    inFile.seekg(0,ios::beg);
    while(inFile.read(reinterpret_cast<char *> (&ac), sizeof(account)))
    {
        if(ac.retaccNo()!=n)
        {
            outFile.write(reinterpret_cast<char *> (&ac), sizeof(account));
        }
    }
    inFile.close();
    outFile.close();
    remove("accdata.dat");
    rename("Temp.dat","accdata.dat");
}

//Function to display all accounts
void dispAll()
{

    ifstream inFile;
    inFile.open("accdata.dat",ios::binary);
    if(!inFile)
    {
        pos(2,12);cout<<"File could not be open !! Press any Key...";
        return;
    }
    col(112);
    cout<<"\n\n\n\t================================================================\n";
    col(124);cout<<"\t Account No.            NAME              Type      Balance     \n";
    col(112);cout<<"\t================================================================\n";
    col(15);
    int i=11;
    while(inFile.read(reinterpret_cast<char *> (&ac), sizeof(account))) {
        ac.accDetails(i);
        i++;
    }
    inFile.close();
}
```

```

//Function to deposit and withdraw amounts
void depwith(int n, int option)
{
    int amt;
    bool found=false;

    fstream File;
    File.open("accdata.dat", ios::binary|ios::in|ios::out);
    if(!File)
    {
        col(12);pos(2,12);cout<<"\n\tNo record found!";col(15);
        return;
    }
    while(!File.eof() && found==false)
    {
        File.read(reinterpret_cast<char *> (&ac), sizeof(account));
        if(ac.retaccNo()==n)
        {
            ac.show();
            if(option==1)
            {
                col(112);cout<<"\n\n\t DEPOSIT AMOUNT ";col(15);
                cout<<"\n\n\tEnter Amount: ";
                cin>>amt;
                ac.deposit(amt);
                ac.show();
                    col(121);cout<<"\n\n\t                  ";
                    cout<<"\n\t  RECORD UPDATED!  ";
                    cout<<"\n\t                   ";col(15);
            }
            if(option==2)
            {
                col(112);cout<<"\n\n\t WITHDRAW AMOUNT ";col(15);
                cout<<"\n\n\tEnter Amount: ";
                cin>>amt;
                int bal=ac.retaccBalance()-amt;

                if((bal<500 && ac.retaccType()=='S') || (bal<1000 && ac.retaccType()=='C')){

                    col(12);cout<<"\n\tInsufficient balance!\n"
                                <<"\tMaintaning Balance for:\n"
                                <<"\tCurrent Account: Php 1000.00\n"
                                <<"\tSavings Account: Php 500.00\n";col(15);
                    }
                else{
                    ac.withdraw(amt);
                    ac.show();
                    col(124);cout<<"\n\n\t                  ";
                    cout<<"\n\t  RECORD UPDATED!  ";
                    cout<<"\n\t                   ";col(15);
                }
            }
            int pos=(-1)*static_cast<int>(sizeof(ac));
            File.seekp(pos,ios::cur);
            File.write(reinterpret_cast<char *> (&ac), sizeof(account));

            found=true;
           }
         }
    File.close();
    if(found==false){

        col(124);cout<<"\n\n\t                     ";
            cout<<"\n\t  RECORD NOT FOUND!  ";
            cout<<"\n\t                     ";col(15);
        }
}

//Intro Function
void welcome()
{
    col(12);pos(2,3);cout <<currentDateTime();
    col(192);pos(50,8);cout << " W E L C O M E  T O ";
    int k;
    col(205);
        for (int j=10; j<15;j++){
        pos(45,j);
             for (int i=0; i<3; i++){
                cout << " " ;
                Sleep(10);
         }
        }
        pos(48,10); cout<<"    ";
        pos(48,14); cout<<"    ";
        for (int j=10; j<15;j++){
        pos(52,j);
             for (int i=0; i<3; i++){
                cout << " " ;
                Sleep(10);
         }
        }
        for (int j=10; j<15;j++){
        pos(56,j);
             for (int i=0; i<3; i++){
                cout << " " ;
                Sleep(10);
         }
        }
        pos(57,14); cout<<"      ";
        col(240);
        for (int j=10; j<15;j++){
        pos(64,j);
             for (int i=0; i<3; i++){
                cout << " " ;
                Sleep(10);
         }
        }
        pos(65,10);cout<<"      ";
        for (int j=10; j<12;j++){
```

```
        pos(71,j);
             for (int i=0; i<3; i++){
                cout << " " ;
                Sleep(10);
         }
        }
        pos(65,12);
        cout << "         ";
        for (int j=12; j<15;j++){
        pos(71,j);
             for (int i=0; i<3; i++){
                cout << " " ;
                Sleep(10);
         }
        }
        pos(65,14);
        cout << "         ";
        col(240);pos(45,16);cout << "O R A N G E  L I O N  B A N K";


    col(15);
    pos(45,20);system("pause>0");
}

//Function for login system
int login()
{
    int opt, num,loop;
    string adminID, adminPass;
    col(121);
    cout << "\n\n\t\t\t [1] Admin Login"
         << "\n\t\t\t [2] User Login "
         << "\n\n\t\t\t Enter option:  ";
    col(112);
    cin >> opt;
    switch (opt)
    {
        case 1:
        do{
            loop=0;
            col(121);cout << "\n\t\t\tEnter Admin ID: ";
            col(112);cin >> adminID;
            col(121);cout << "\t\t\tEnter Admin Password: ";
            col(112);cin >> adminPass;

            if((adminID=="admin")&&(adminPass=="1234"))
            {
                col(121);
                cout<<"\n\t\t\t                    ";
                cout<<"\n\t\t\t   WELCOME ADMIN!   ";
                cout<<"\n\t\t\t                    \n\n\t\t\t";
                col(15);
                system("pause");
            }
            else
            {
                col(12);
                cout<<"\n\t\t\tIncorrect Admin ID and Password    \n\t\t\t";
                col(15);
                system("pause");
                loop=1;

            }

        }while(loop==1);
            break;
        case 2:
            col(112);cout << "\t\t\tEnter Account No.: ";
            cin >> num;
            usernum = num;
            col(15);
            matchAccount(num);
            break;
        default: col(12);cout <<"\n\t\t\tInvalid Option! Goodbye!";col(15);
                    return 0; system("pause");col(15);
    }
    return opt;
}

void matchAccount(int n)
{int loop;
    do
    {
    loop = 1;
    int pin;
    bool found1=false,found2=false;
    fstream File;
    File.open("accdata.dat",ios::binary|ios::in|ios::out);

    while(!File.eof() && found1==false)
    {
        File.read(reinterpret_cast<char *> (&ac), sizeof(account));
        if(ac.retaccNo()==n)
        {found1=true;
        }

    }
    File.close();
    if(found1==true)
    {
        col(112);cout <<"\t\t\tEnter Account PIN: ";
        cin>>pin;col(15);


            while(!File.eof() && found2==false)
            {

            File.read(reinterpret_cast<char *> (&ac), sizeof(account));
            if(ac.retaccPin()==pin)
            {
```

```

            found2=true;}
            }
            File.close();
            if((found1==true)&&(found2==true)){

            col(121);
                cout<<"\n\t\t\t                   ";
                cout<<"\n\t\t\t   WELCOME USER!   ";
                cout<<"\n\t\t\t                   \n\n\t\t\t";
                col(15);
            system("pause");
            loop = 0;
        }
            else {
                col(12);
                cout<<"\t\t\tIncorrect PIN!\n\t\t\t";
                col(15);
                system("pause");
            }
    }
    else
    {
        col(12);cout <<"\n\t\t\tAccount does not exist!";
        col(112);cout << "\n\n\t\t\tEnter Account No.: ";
        cin >> n;col(15);
    }


}while(loop ==1);

}

/*****************************************************/
//          MAIN PROGRAM                             //
/****************************************************/

int main()
{
    int user;
    welcome();
    user = login();
    string ch;
    int num;
    if(user==1)
{
    do
    {

        system("cls");
        logoandtime();
        col(112);
        pos(39,5);cout<<"               ";
        pos(39,6);cout<<"   MAIN MENU   ";
        pos(39,7);cout<<"               ";
        pos(6,10);cout<<"                         ";
        pos(6,11);cout<<"   [1] CREATE ACCOUNT    ";
        pos(6,12);cout<<"                         ";
        pos(6,14);cout<<"                         ";
        pos(6,15);cout<<"   [2] DEPOSIT           ";
        pos(6,16);cout<<"                         ";
        pos(6,18);cout<<"                         ";
        pos(6,19);cout<<"   [3] WITHDRAW          ";
        pos(6,20);cout<<"                         ";
        pos(6,22);cout<<"                         ";
        pos(6,23);cout<<"   [4] BALANCE INQUIRY   ";
        pos(6,24);cout<<"                         ";
        pos(35,10);cout<<"                         ";
        pos(35,11);cout<<"   [5] VIEW ALL ACCOUNTS ";
        pos(35,12);cout<<"                         ";
        pos(35,14);cout<<"                         ";
        pos(35,15);cout<<"   [6] CLOSE ACCOUNT     ";
        pos(35,16);cout<<"                         ";
        pos(35,18);cout<<"                         ";
        pos(35,19);cout<<"   [7] MODIFY ACCOUNT    ";
        pos(35,20);cout<<"                         ";
        pos(35,22);cout<<"                         ";
        pos(35,23);cout<<"   [8] EXIT              ";
        pos(35,24);cout<<"                         ";
        col(124);
        pos(63,17);cout<<"Enter Option (1-8): ";
        col(112);cin >>ch; col(15);

        system("cls");

        if (ch=="1")
            writeAccount();

        else if(ch=="2"){
            logoandtime();
            cout<<"\n\n\tEnter Account No. : "; cin>>num;
            depwith(num, 1);
        }
        else if(ch=="3"){
            logoandtime();
            cout<<"\n\n\tEnter Account No. : "; cin>>num;
            depwith(num, 2);
            }
        else if(ch=="4"){
            logoandtime();
            cout<<"\n\n\tEnter Account No. : "; cin>>num;
            displayAccount(num);
        }
        else if(ch=="5"){
        logoandtime();
            dispAll();
        }
        else if(ch=="6"){
        logoandtime();
            cout<<"\n\n\tEnter Account No. : "; cin>>num;
            deleteAccount(num);
        }
         else if(ch=="7"){
        logoandtime();
            cout<<"\n\n\tEnter Account No. : "; cin>>num;
            modifyAccount(num,1);
        }
         else if(ch=="8"){
            logoandtime();
            col(15);
            cout<<"\n\n\tThank you! Have a nice day!\n";col(15);
        }
         else {
         logoandtime();
                col(12); cout<<"\n\n\tINVALID CHOICE\n";col(15);
        }
        cin.ignore();
        cin.get();
    }while(ch!="8");
}
    else if(user == 2)
    {
            do
    {

        system("cls");
        logoandtime();
        col(112);
        pos(39,5);cout<<"              ";
        pos(39,6);cout<<"  MAIN MENU   ";
        pos(39,7);cout<<"              ";
        pos(6,10);cout<<"                        ";
        pos(6,11);cout<<"  [1] DEPOSIT           ";
        pos(6,12);cout<<"                        ";
        pos(6,14);cout<<"                        ";
        pos(6,15);cout<<"  [2] WITHDRAW          ";
        pos(6,16);cout<<"                        ";
        pos(6,18);cout<<"                        ";
        pos(6,19);cout<<"  [3] BALANCE INQUIRY   ";
        pos(6,20);cout<<"                        ";
        pos(6,22);cout<<"                        ";
        pos(6,23);cout<<"  [4] CHANGE PIN        ";
        pos(6,24);cout<<"                        ";
        pos(6,26);cout<<"                        ";
        pos(6,27);cout<<"  [5] EXIT              ";
        pos(6,28);cout<<"                        ";
        col(124);
        pos(39,17);cout<<"Enter Option (1-5): ";
        col(112);cin >>ch; col(15);

        system("cls");


        if(ch=="1"){

            logoandtime();
            num=usernum;
            depwith(num, 1);
        }
        else if(ch=="2"){
        logoandtime();
            num=usernum;
            depwith(num, 2);
        }
        else if(ch=="3"){
        logoandtime();
            num=usernum;
            displayAccount(num);
        }
        else if(ch=="4"){
        logoandtime();
            num=usernum;
            modifyAccount(num,2);}
        else if (ch=="5"){
        logoandtime();
            col(15);
            cout<<"\n\n\tThank you! Have a nice day!\n";
            ch = '8';
        }
        else {
        logoandtime();
                col(12); cout<<"\n\n\tINVALID CHOICE\n";col(15);
        }
        cin.ignore();
        cin.get();
    }while(ch!="8");
    }


}
////////////////////////////END OF PROGRAM///////////////////////////////////
//        YEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEESSSSS
```