“Heaven’s Light is Our Guide”

Department of Computer Science & Engineering

RAJSHAHI UNIVERSITY OF ENGINEERING & TECHNOLOGY

Lab Report-04

Submitted By:

Name: Khandoker Sefayet Alam
Roll:2003121
Department: Computer Science & Engineering
Section-C
Session:2020-21
Course code: CSE 1204

Submitted to:

SUHRID SHAKHAR GHOSH
Assistant Professor
Department of Computer Science & Engineering, RUET

[module-01]

Module 1: (for Week 1)

Problem Statement: Write a C++ program using class and objects to automate the banking process with following constraints:

i. Only admin can open new account or close an old account

ii. An account holder can debit/credit money

iii. An account can't be deleted if money exists in the account

iv. The program should be following menu operated

v. The private data members of an account holder's class are "Account No", "Name", "Age" and "Balance".

vi. Design member methods for individual operation

vii. Produce appropriate message when required

a.The template of the class is

```
class Account{
//private data members
//public methods
};
```

b.The menu is :

```
********** Main Menu ********

   1. Open New Account
   2. Close Old Account
   3. Deposit Money
   4. Withdraw Money
   5. Check Balance
   6. Exit
      Enter Your option(1-6):_
```

c.2Few Snapshots:

1. Deposit Money

```
Enter your option(1-6):3 <enter>
Enter Account Number:12678 <enter>
Enter Amount: 5000 <enter>
Successfully deposited
Press any key go to main menu.....
```

Solution:

```
#include<iostream>

using namespace std;

int idx=0;
int trid=0;
bool vis[1010];
class Account{
private:
   int accountno;
   int age;
   string name;
   int balance;
public:

   ///setters
   void setaccountno(int a){
   accountno=a;
   }
   void setage(int a){
   age=a;
   }
   void setbalance(int a){
   balance=a;
   }
   void setname(string a){
   name=a;
   }
   ///getters
   int getaccountno(){
   return accountno;
   }
   int getage(){
   return age;
   }
```

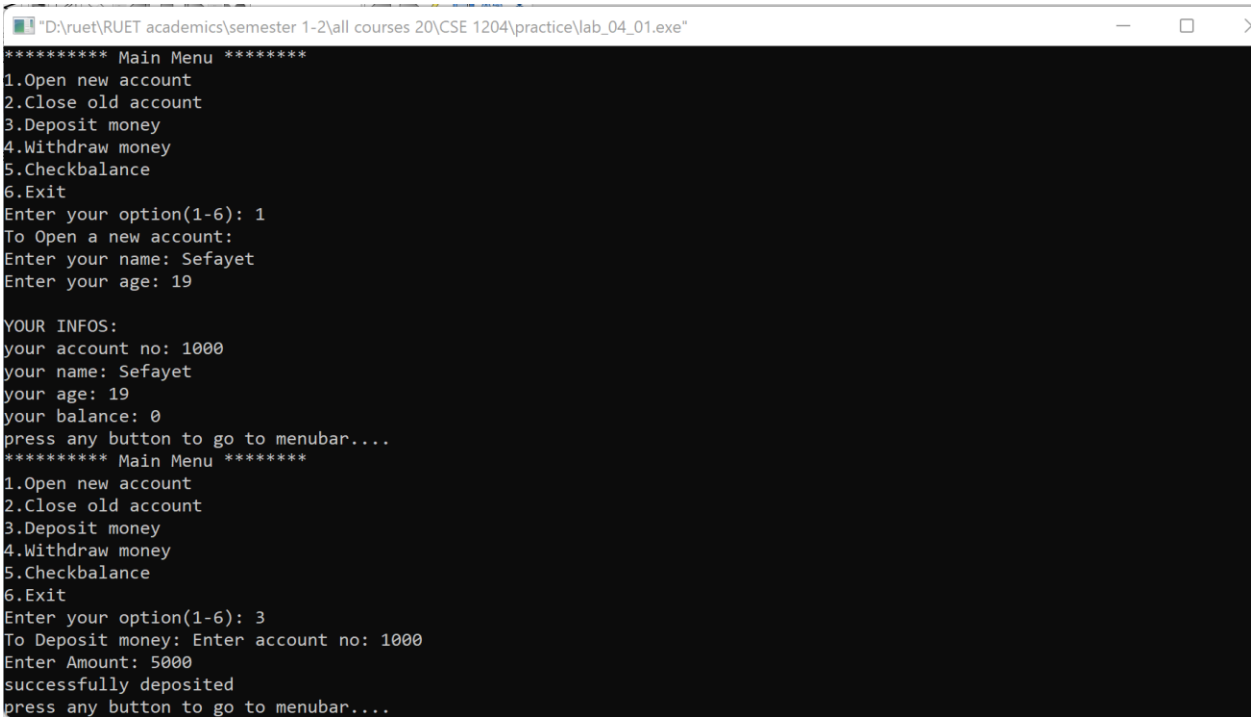
```
   int getbalance(){
   return balance;
   }
   string getname(){
   return name;
   }
   ///deposit and withdraw
   void balanceincr(int x){
   balance+=x;
   }
   void balancedec(int x){
   balance-=x;
   }

};
Account acc[1000];
void Enterinfos(){
      string n;
      cout<<"To Open a new account: "<<endl;
      cout<<"Enter your name: ";
      cin>>n;
      int k;
      cout<<"Enter your age: ";
      cin>>k;
      acc[idx].setaccountno(1000+idx);
      acc[idx].setage(k);
      acc[idx].setname(n);
      acc[idx].setbalance(0);
      vis[idx]=1;
      cout<<endl;
      cout<<"YOUR INFOS: "<<endl;
      cout<<"your account no: "<<acc[idx].getaccountno()<<endl;
      cout<<"your name: "<<acc[idx].getname()<<endl;
      cout<<"your age: "<<acc[idx].getage()<<endl;
      cout<<"your balance: "<<acc[idx].getbalance()<<endl;

}
```

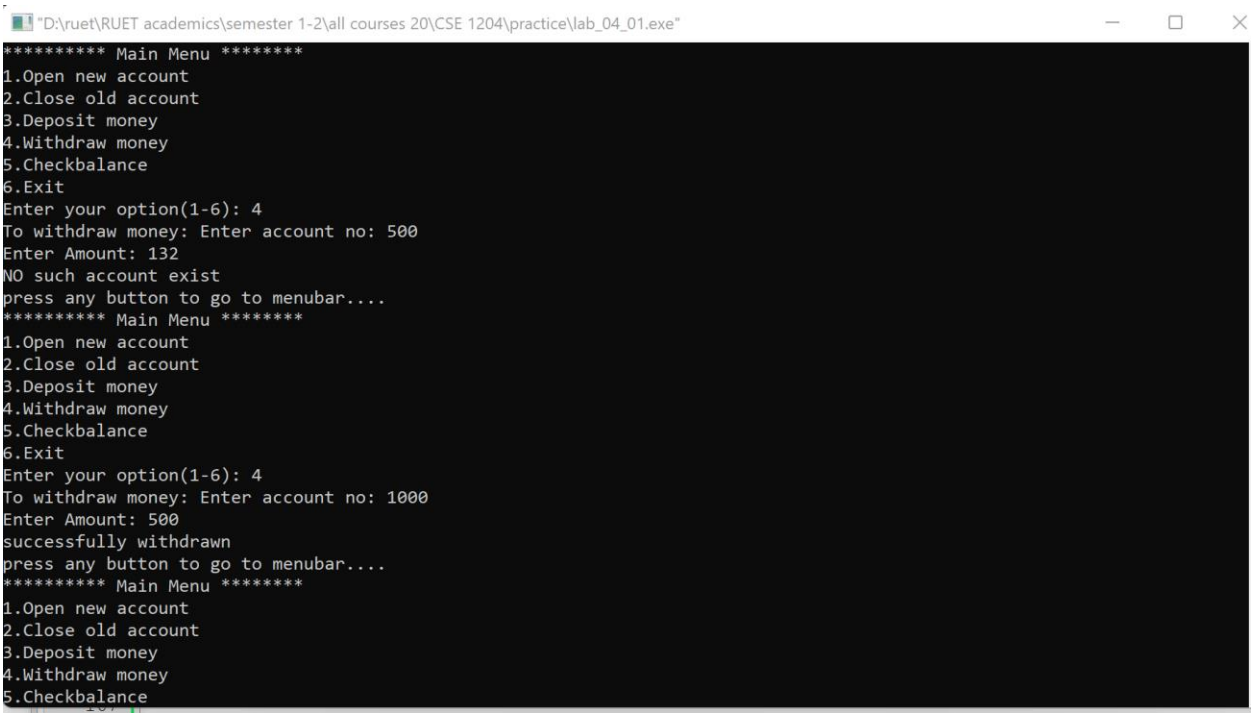
```
void closingacc(){
	int num;
	cout<<"Enter your account no to close the account: ";
	cin>>num;
	acc[num-1000].setaccountno(num);
	acc[num-1000].setage(0);
	acc[num-1000].setname("deleted account");
	acc[num-1000].setbalance(0);
	vis[idx]=0;
	cout<<"Account is deleted"<<endl;
}
void deposit(){
	int num2,num3;
	cout<<"To Deposit money: ";

	cout<<"Enter account no: ";
	cin>>num2;
	cout<<"Enter Amount: ";
	cin>>num3;
	if(vis[num2-1000 ]){acc[num2-1000].balanceincr(num3);
	cout<<"successfully deposited"<<endl;}
	else{
	cout<<"NO such account exist"<<endl;
	}


}
void checkbalance(){
	int num2;
	cout<<"Enter account no: ";
	cin>>num2;
	if(vis[num2-1000]){
	cout<<"Your account balance= "<<acc[num2-1000].getbalance()<<endl;
	}
	else{
	   cout<<"NO such account exist"<<endl;
	}
```

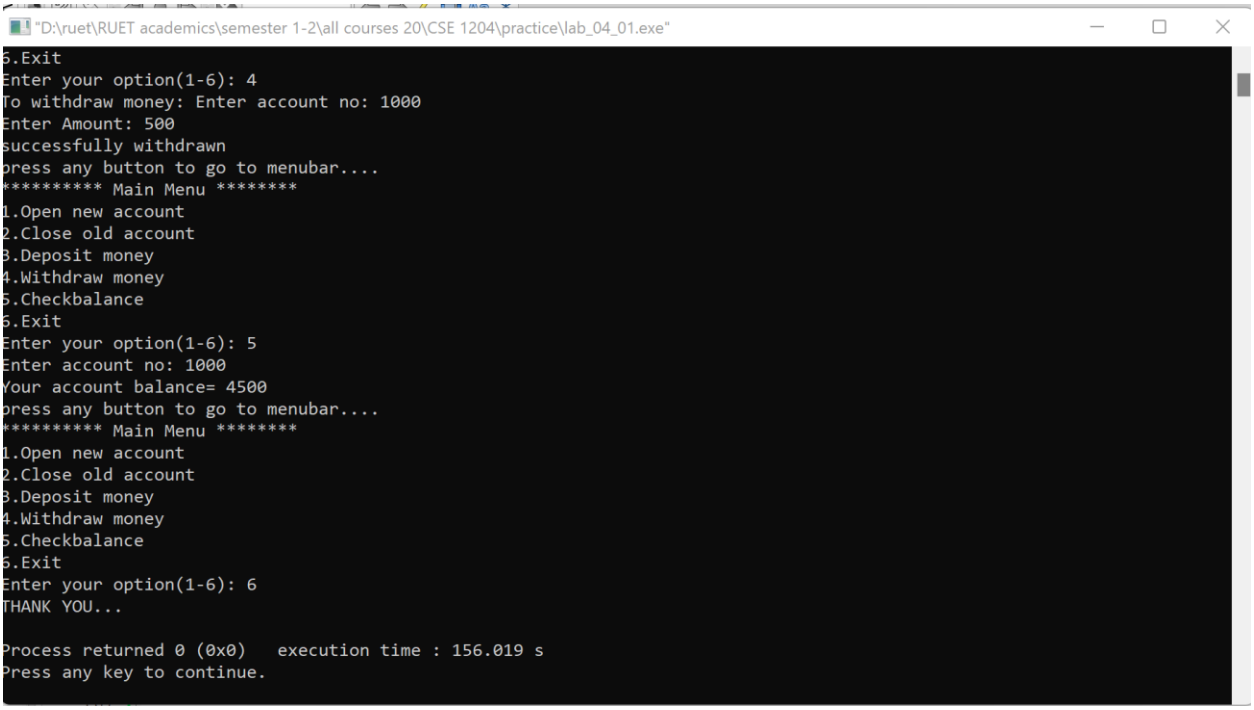
```
}
void withdraw(){
    int num2,num3;
      cout<<"To withdraw money: ";

      cout<<"Enter account no: ";
      cin>>num2;
      cout<<"Enter Amount: ";
      cin>>num3;
      if(vis[num2-1000]){
      if(acc[num2-1000].getbalance()>=num3) {acc[num2-
1000].balancedec(num3);
      cout<<"successfully withdrawn"<<endl;}
      else{
         cout<<"Not enough balance!"<<endl;
      }
      }
       else{
         cout<<"NO such account exist"<<endl;
      }

}
int main(){

   while(1){
      bool f=0;

   cout<<"1.Open new account\n2.Close old account\n3.Deposit
money\n4.Withdraw money\n5.Checkbalance\n6.Exit"<<endl;
   int choice;
   cout<<"Enter your option(1-6): ";
   cin>>choice;
   switch(choice){
   case 1:
     Enterinfos();
      break;
   case 2:
```

```
    closingacc();
  break;
  case 3:
    deposit();
  break;
  case 4:
    withdraw();
  break;
  case 5:
    checkbalance();

  break;
  case 6:
    cout<<"THANK YOU..."<<endl;
    f=1;
  break;
  default:
    cout<<"Enter a valid choice"<<endl;
  }
  if(f) break;
  }
```

OUTPUT:







[module:02]

Task-01: **Use the following class Test with the data members and methods**

```
class Test{
private:
```

```
Data Member x;
Data Member y;
Data Member z;
public:
//write methods
};
```

Now extend the program do the following
i) Initialize private data members x and y to 0 when empty constructor is called
ii) Initialize private data members x and y using parameterized constructor is called
iii) Initialize private data members x and y from another object using copy constructor
iv) The data member z keeps track of total objects created
v) Write a method to initialize x and y
vi) Write a method to display data member z only
vii) Write a method to display x,y and z where their values can't be changed
viii) Create five objects
ix) Find the sum of x
x) Find the object number whose y value is maximum

Solution:

```
#include<iostream>


using namespace std;


class Test{
private:
   int x;
   int y;
   static int z;
public:


   Test(){
   x=0;
   y=0;
```

```
z++;
}

Test(int a,int b){
x=a;
y=b;
z++;
}

Test(Test &r){
x=r.x;
y=r.y;
z++;
}
int getx(){
return x;
}
int gety(){
return y;
}
void setxy(int a,int b){
x=a;
y=b;
}
static void getz(){
cout<<"z= "<<z<<endl;
```

```
    }
    void display() const{
    //x++,y++,z++
    cout<<"x= "<<x<<endl;
    cout<<"y= "<<y<<endl;
    cout<<"z= "<<z<<endl;
    }

};

int Test::z=0;

int main(){
    Test obj[5];//creating five objects
    int sum=0;
    int maxmy=0;
    int objno;
    int k,l;
    for(int i=0;i<5;i++){
        cout<<"for obj no:"<<i+1<<" enter the value of x and y: ";
        cin>>k>>l;
        obj[i].setxy(k,l);
        // initializing x and y using method

        sum+=obj[i].getx();
        if(obj[i].gety() > maxmy) {objno=i+1;maxmy=obj[i].gety();}
```

```
    }
    cout<<"Sum of x of 5 objects= "<<sum<<endl;
    cout<<"maxmy= "<<maxmy<<" and the obj no is: "<<objno<<endl;
   cout<<"Number of objects created ";
   obj[0].getz();
Test newobj(5,6);//using the parameterized constructor
   Test obj2(newobj);//using copy constructor
    cout<<"new obj: "<<endl;
    newobj.display();

    cout<<"Copied obj: "<<endl;
    obj2.display();
}
```