

Shlok Agrawal
0801CS211081
PP-final project
Bank Management System

1 Overview

This is a bank management system in which a user can perform actions like: opening an account, depositing/withdrawing money from the account, viewing the account details.

2 Code Description and Functions

The code is divided in to a main function and some spcific functions for every action. The main function calls the other functions. The following functions were used to make the project:

2.1 Open Account

This function can be used to open a new account. The user can provide details like name, address, account type (savings/current), and initial deposit amount.

2.2 Deposit money

This function can be used to deposit a sum of money to an existing account. The amount to be deposited is taken as input and the balnace is updated accordinlgy.

2.3 Withdraw money

This function can be used to withdraw a sum of money from an existing account. The amount to be withdrawn is taken as input and the balnace is updated accordinlgy.

2.4 Display Account

This function can be used to view the details like name, address, account type (savings/current) and account balance of an existing account.

3 Code in c++

```
#include <iostream>
#include <string>
#include<bits/stdc++.h>
```

```
using namespace std;

class bank
{   char name[100], add[100],t;
    int balance;
    public:
        void open_account();
        void deposit_money();
        void withdraw_money();
        void display_account();

};

void bank::open_account()
{
    cout<<"enter name\n";
    cin.ignore();
    cin.getline(name,100);

    cout<<"enter address\n";
    cin.ignore();
    cin.getline(add,100);

    cout<<"account type saving (s) or current (c)\n";
    cin>>t;

    cout<<"deposit amount ";
    cin>>balance;
    cout<<"account created";

}

void bank::deposit_money()
{   int amount;
    cout<<"enetr amount to deposit\n";
    cin>>amount;
    balance+=amount;
    cout<<"total balance\n"<<balance;

}

void bank::display_account()
{
    cout<<"Name:"<<name<<"\n";
    cout<<"Address:"<<add<<"\n";
    cout<<"account type"<<t<<"\n";
    cout<<"Balance:"<<balance<<"\n";
}
```

```
void bank::withdraw_money()
{
    float amount = 0;

    cout<<"current balance"<<balance<<"\n";
    while (amount<balance)
    {
        cout<<"enter withdrawal amount:";
        cin>>amount;
        balance-=amount;
    }
    cout<<"balance is"<<balance<<"\n";
}
```

```
int main()
{
    int menu,x;
    bank obj;
    do{
        cout<<"1. open account\n";
        cout<<"2. deposit money\n";
        cout<<"3. withdraw money\n";
        cout<<"4. display account\n";
        cout<<"5. Exit\n";
        cout<<"choose an option ";
        cin>>menu;
        switch (menu)
        {
            case 1:
                obj.open_account();
                break;
            case 2:
                obj.deposit_money();
                break;
            case 3:
                obj.withdraw_money();
                break;
            case 4:
                obj.display_account();
                break;
            case 5: exit (1);
                break;
            default:
                cout<<"enter suitable number\n";
        }
    }
```

```
    cout<<"\n if you want to select an option press :: 1\n";  
    cout<<"if you want to exit press::0 ";  
    cin>>x;  
    if(x==2)  
        exit(0);  
    } while (x==1);  
}
```

4 Execution and Compilation

```

amirochates@amirochates-Vostro1510: ~/Desktop/Sem 3/bank-pp$ g++ -pg bank.cpp
amirochates@amirochates-Vostro1510:~/Desktop/Sem 3/bank-pp$ ./a.out
1. open account
2. deposit money
3. withdraw money
4. display account
5. Exit
choose an option 1
enter name
Shlok
enter address
12, absc
account type saving (s) or current (c)
s
deposit amount 1200
account created
if you want to select an option press :: 1
if you want to exit press::0 1
1. open account
2. deposit money
3. withdraw money
4. display account
5. Exit
choose an option 2
enetr amount to deposit
1200
total balance
2400
if you want to select an option press :: 1
if you want to exit press::0 1
1. open account
2. deposit money
3. withdraw money
4. display account
5. Exit
choose an option 3
current balance2400
enter withdrawal amount:1200
balance is1200

amirochates@amirochates-Vostro1510:~/Desktop/Sem 3/bank-pp$
1. open account
2. deposit money
3. withdraw money
4. display account
5. Exit
choose an option 3
current balance2400
enter withdrawal amount:1200
balance is1200

if you want to select an option press :: 1
if you want to exit press::0 1
1. open account
2. deposit money
3. withdraw money
4. display account
5. Exit
choose an option 4
Name:Shlok
Address:2, absc
account types
Balance:1200

if you want to select an option press :: 1
if you want to exit press::0 1
1. open account
2. deposit money
3. withdraw money
4. display account
5. Exit
choose an option 5
amirochates@amirochates-Vostro1510:~/Desktop/Sem 3/bank-pp$ gmon.out
gmon.out: command not found
amirochates@amirochates-Vostro1510:~/Desktop/Sem 3/bank-pp$ ./gmon.out
bash: ./gmon.out: Permission denied
amirochates@amirochates-Vostro1510:~/Desktop/Sem 3/bank-pp$ sudo ./gmon.out
[sudo] password for amirochates:
sudo: ./gmon.out: command not found
amirochates@amirochates-Vostro1510:~/Desktop/Sem 3/bank-pp$ gprof bank > bank.output
amirochates@amirochates-Vostro1510:~/Desktop/Sem 3/bank-pp$

```

5 Profiling

The screenshot shows a text editor window titled "bank.output" with the following content:

```

1 Flat profile:
2
3 Each sample counts as 0.01 seconds.
4 no time accumulated
5
6 % cumulative self self total
7 time seconds seconds calls Ts/call Ts/call name
8 0.00 0.00 0.00 1 0.00 0.00 _GLOBAL_sub_I_ZN4bank12open_accountEv
9 0.00 0.00 0.00 1 0.00 0.00 _static_initialization_and_destruction_0(int, int)
10 0.00 0.00 0.00 1 0.00 0.00 bank::open_account()
11 0.00 0.00 0.00 1 0.00 0.00 bank::deposit_money()
12 0.00 0.00 0.00 1 0.00 0.00 bank::withdraw_money()
13 0.00 0.00 0.00 1 0.00 0.00 bank::display_account()
14
15 % the percentage of the total running time of the
16 time program used by this function.
17
18 cumulative a running sum of the number of seconds accounted
19 seconds for by this function and those listed above it.
20
21 self the number of seconds accounted for by this
22 seconds function alone. This is the major sort for this
23 listing.
24
25 calls the number of times this function was invoked, if
26 this function is profiled, else blank.
27
28 self the average number of milliseconds spent in this
29 ms/call function per call, if this function is profiled,
30 else blank.
31
32 total the average number of milliseconds spent in this
33 ms/call function and its descendents per call, if this
34 function is profiled, else blank.
35
36 name the name of the function. This is the minor sort
37 for this listing. The index shows the location of
38 the function in the gprof listing. If the index is
39 in parenthesis it shows where it would appear in

```

Below the flat profile, there is a call graph section:

```

45 are permitted in any medium without royalty provided the copyright
46 notice and this notice are preserved.
47
48 Call graph (explanation follows)
49
50
51 granularity: each sample hit covers 2 byte(s) no time propagated
52
53 index % time self children called name
54
55 [8] 0.0 0.00 0.00 1/1 _GLOBAL_sub_I_ZN4bank12open_accountEv [8]
56 0.00 0.00 1/1 _static_initialization_and_destruction_0(int, int) [9]
57 -----
58 0.00 0.00 1/1 _GLOBAL_sub_I_ZN4bank12open_accountEv [8]
59 [9] 0.0 0.00 0.00 1 _static_initialization_and_destruction_0(int, int) [9]
60 -----
61 0.00 0.00 1/1 main [6]
62 [10] 0.0 0.00 0.00 1 bank::open_account() [10]
63 -----
64 0.00 0.00 1/1 main [6]
65 [11] 0.0 0.00 0.00 1 bank::deposit_money() [11]
66 -----
67 0.00 0.00 1/1 main [6]
68 [12] 0.0 0.00 0.00 1 bank::withdraw_money() [12]
69 -----
70 0.00 0.00 1/1 main [6]
71 [13] 0.0 0.00 0.00 1 bank::display_account() [13]
72 -----
73
74 This table describes the call tree of the program, and was sorted by
75 the total amount of time spent in each function and its children.
76
77 Each entry in this table consists of several lines. The line with the
78 index number at the left hand margin lists the current function.
79 The lines above it list the functions that called this function,
80 and the lines below it list the functions this one called.
81 This line lists:
82 index A unique number given to each element of the table.
83 Index numbers are sorted numerically.

```

6 Debugging

The bug found in the deposit money function was that the balance was not updating properly after the new amount deposited was added. The error was a missing + sign in the balance updation line which was later fixed.

```

amitrochates@amitrochates-Vostro1510: ~/Desktop/Sem 3/bank-pp
amitrochates@amitrochates-Vostro1510:~/Desktop/Sem 3/bank-pp$ g++ -g bank.cpp -g
amitrochates@amitrochates-Vostro1510:~/Desktop/Sem 3/bank-pp$ gdb ./a.out
GNU gdb (Ubuntu 9.2-0ubuntu1~20.04.1) 9.2
Copyright (C) 2020 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./a.out...
(gdb) b 40
Breakpoint 1 at 0x13df: file bank.cpp, line 40.
(gdb) run
Starting program: /home/amitrochates/Desktop/Sem 3/bank-pp/a.out
1. open account
2. deposit money
3. withdraw money
4. display account
5. Exit
choose an option 1
enter name
shlok
enter address
12, abc
account type saving (s) or current (c)
s
deposit amount 1000
account created
if you want to select an option press :: 1
if you want to exit press::0 1
:::
1. open account
2. deposit money

```

```
Activities Terminal Nov 20 10:27 PM
amitrochates@amitrochates-Vostro1510: ~/Desktop/Sem 3/bank-pp

Starting program: /home/amitrochates/Desktop/Sem 3/bank-pp/a.out
1. open account
2. deposit money
3. withdraw money
4. display account
5. Exit
choose an option 1
enter name
shlok
enter address
12, abc
account type saving (s) or current (c)
s
deposit amount 1000
account created
if you want to select an option press :: 1
if you want to exit press::0 1
1. open account
2. deposit money
3. withdraw money
4. display account
5. Exit
choose an option 2
enetr amount to deposit
200

Breakpoint 1, bank::deposit_money (this=0x7ffffffde40) at bank.cpp:40
40      balance=amount;
(gdb) step
42      cout<<"total balance\n"<<balance;
(gdb) next
total balance
45      }
(gdb) quit
A debugging session is active.

Inferior 1 [process 22747] wlll be killed.

Quit anyway? (y or n) y
amitrochates@amitrochates-Vostro1510:~/Desktop/Sem 3/bank-pp$
```