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 accordingly.

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 The amount to be withdrawn is taken as input and the balnace is updated accordingly.

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++

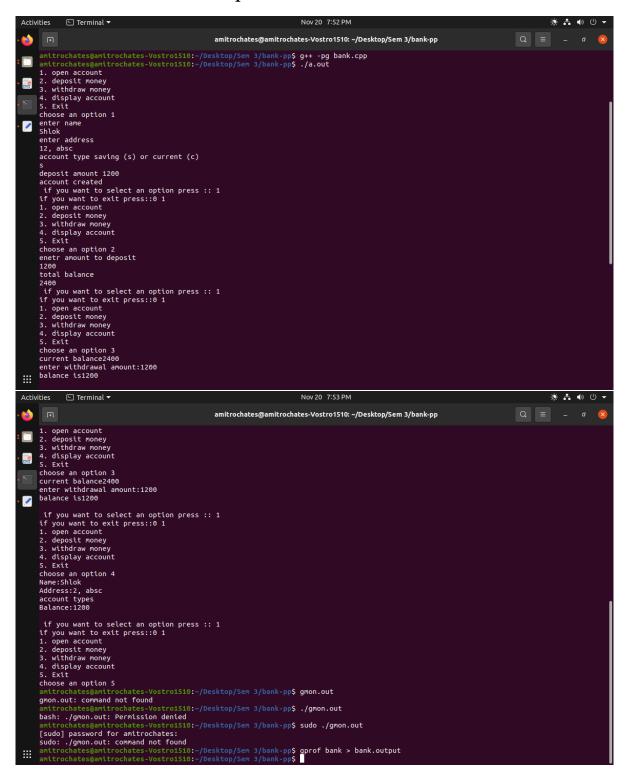
#include <iostream>
#include <string>
#include<bits/stdc++.h>

```
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";</pre>
    cin.ignore();
    cin.getline(name,100);
    cout<<"enter address\n";</pre>
    cin.ignore();
    cin.getline(add,100);
    cout<<"account type saving (s) or current (c)\n";</pre>
    cin>>t;
    cout<<"deposit amount ";</pre>
    cin>>balance;
    cout<<"account created";</pre>
}
    void bank::deposit_money()
        int amount;
        cout<<"enetr amount to deposit\n";</pre>
        cin>>amount;
        balance+=amount;
        cout<<"total balance\n"<<balance;</pre>
    }
void bank::display_account()
    cout<<"Name:"<<name<<"\n";</pre>
    cout<<"Address:"<<add<<"\n";</pre>
    cout<<"account type"<<t<<"\n";</pre>
    cout<<"Balance:"<<balance<<"\n";</pre>
}
```

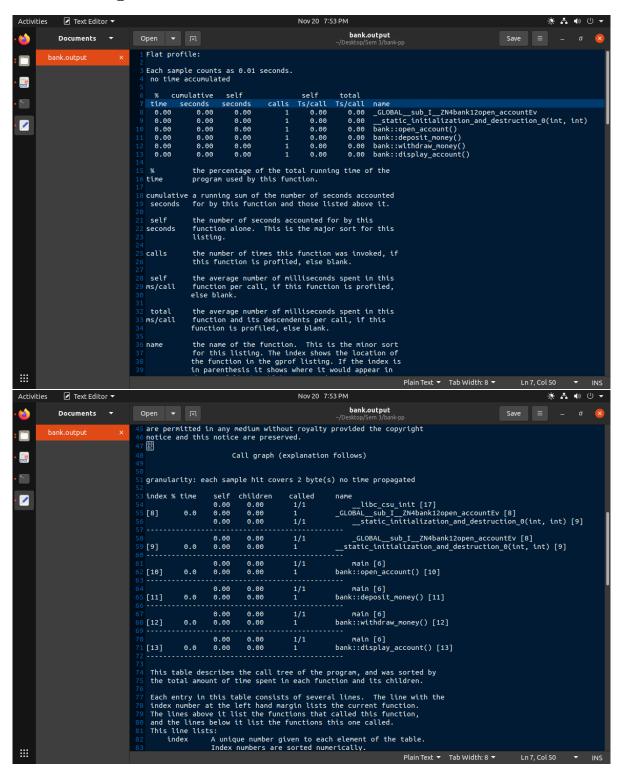
```
void bank::withdraw_money()
    float amount = 0;
    cout<<"current balance"<<balance<<"\n";</pre>
    while (amount<balance)</pre>
    {
    cout<<"enter withdrawal amount:";</pre>
    cin>>amount;
    balance-=amount;
    }
    cout<<"balance is"<<balance<<"\n";</pre>
}
int main()
{
    int menu,x;
    bank obj;
    do{
    cout << "1. open account \n";
    cout<<"2. deposit money\n";</pre>
    cout<<"3. withdraw money\n";</pre>
    cout<<"4. display account\n";</pre>
    cout<<"5. Exit\n";</pre>
    cout<<"choose an option ";</pre>
    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";</pre>
    }
```

```
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



5 Profiling



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.

