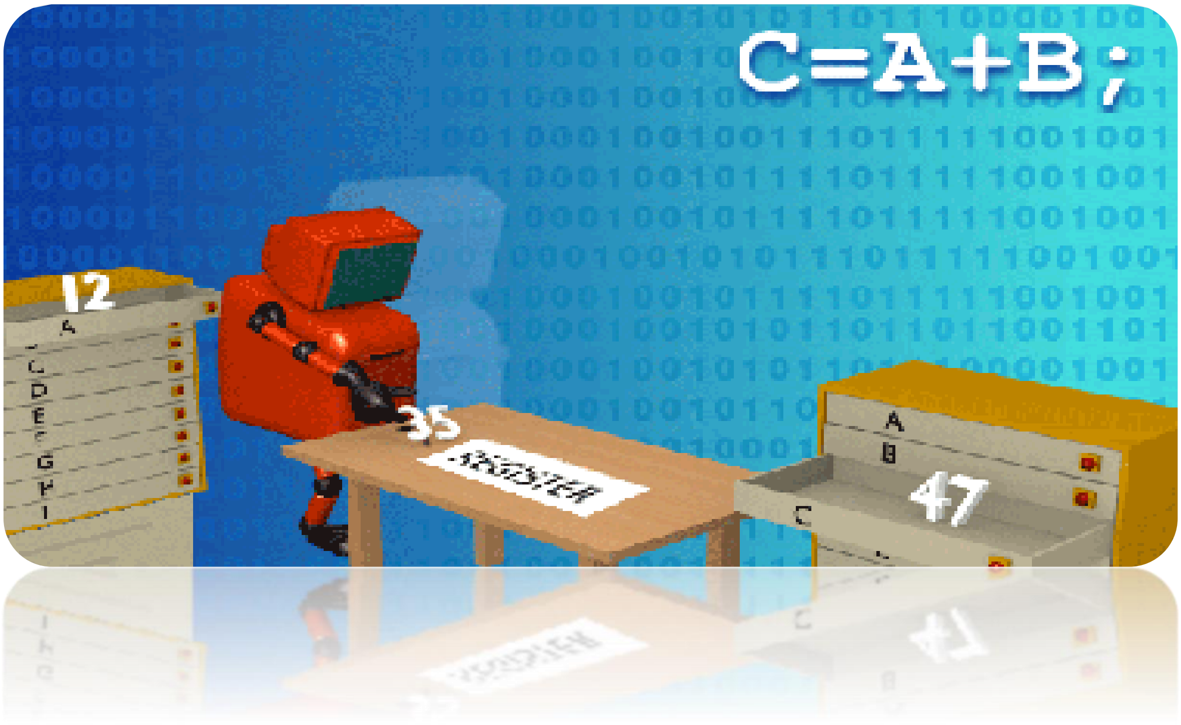
BASICS OF PROGRAMMIING 2

FINAL PROJECT REPORT

BANK MANGEMENT SYSTEM WITH DIFFERENT TYPE OF ACCOUNTS



NAME:

MUNEEB ALI

NAPTUNE CODE:

ZGOKE2

SUBMITTED TO:

MUHAMMAD SALEEM

1 Table of Contents

PICTURE INDEX

Introduction to Project:

Bank is an Economical institution that accepts money from the public and creates account and lending activities can be performed either directly or indirectly. It receives money from those clients who want save in the form of deposits and due to their importance in the financial stability of the country, bank is highly regulated in most countries.

1.1 Motivation:

In this project I made data base on the structure of an old bank in my home country. Terms and conditions are applied on the structure of this bank. I kept in mind the basic structure of this bank in my mind while making this project.

1.2 Problem Statement:

Bank management system in c++ by using the Techniques of object-oriented programming with file handling, exception handling and Dynamic memory allocation.

1.3 Task Description:

This is project make to control the banking system by the banking staff. The banker can perform the operation on it according to the need of his clients. This project solves the problem of data management. Make three basic types of Bank Accounts, Delete the account, modify the account, tell the amount of money stored in the particular account, printing all the account in the data base with their credential, printing the credentials of particular account, finding the exchange rate from different other currencies to Hungarian forints and calculate the amount in the Hungarian forints if we have money in other currencies, it also tell the amount of interest rate per transaction in different accounts.

The interest rate varies from the account type to different account type. When we make account, we also have to deposit certain amount of money which also varies from account to account. If we want to deposit the amount to money less then recommended the system denies the account. Before make the account there is a certain legal condition with documents which the client have to fulfill otherwise, he is denied by the making new account. He is only allowed to make new account when he fulfils these conditions which are again varies from the one account type to other account type. Before deleting an account, the account number and the current account type should be known before otherwise the system cannot be able to delete the account. There is a same condition for modifying account like you have to enter the account number currently and account type. If in the case you enter the wrong credentials its means that this doesn’t exist in the system. System will reply you the account doesn’t exists.

2 Program Analysis/Design solution:

2.1 How to solve it?

To design the solution of the problem first all I used the basic c++ classes. I make the one base class where is stored the date in the form of my variables and then also defined the class functions which was common in all the children classes. I also make the constructors and destructors to allocate the memory dynamically and then also destructor to return the memory to the system after using all of these working. I also define the functions as virtual

Which will be used in the child classes. Then I make the three child classes to for the saving account, Current account, universe student account. Then I accessed the function of the main class into the child class. Define their definitions in their respective cpp files .Then I make another file to call these all function in my main function so that user can interact with these functions. I applied different conditions to take input from the user, different requirements for different types of an Account. Functions like making modifying account, show account details, function for the returning the interest rate are defined in the class header files and then defined their respective files.

Then I add one file where I call these functions that I has defined in the child classes. This file has more than ten functions that are later on called the main menu. First of a function for account making is made. In which users are asked to first of all enter the type of an account which the client want to open, this is very important because at every account type there are different rules are applicable. After this there is a list of requirements that the user has to fulfil for opening accounts, if clients has these required things then user open the account and program ask to enter the current information about the client like his name, his father name, his address, Id card number and the amount of money that initial that has to add in the account in the time of account opening . If enter the money system will not accept until you enter valid amount. After all of that you your account is successfully created and all the date that is assign to variables of the classes will be store in the file.

Then I make function to print all the data from the date base to screen .It first of all open the files for every account and the store data in the object call the function of the class to print all the data .It print all the data until the end of file.

Then I make the function for the for printing the credentials of particular account. First of all, it will ask the user to enter the account number and then account type. It will compare will the data present in the files. If the account number matches it will print data of this account number.

Then I made a function for deleting an account from the data base for this make the one temporary file .User of the project is asked to enter the account number and the account type. I took all the data except with the account number entered by the user into the temporary. Also make the function for balance inquiry that take the account number and print the amount of money stored in file for this particular account.

Then I make two function to advance my project to print the latest exchange rate and then calculating exchanged money from different account to Hungarian forints.

2.2 Flowchart:

- Main classes:

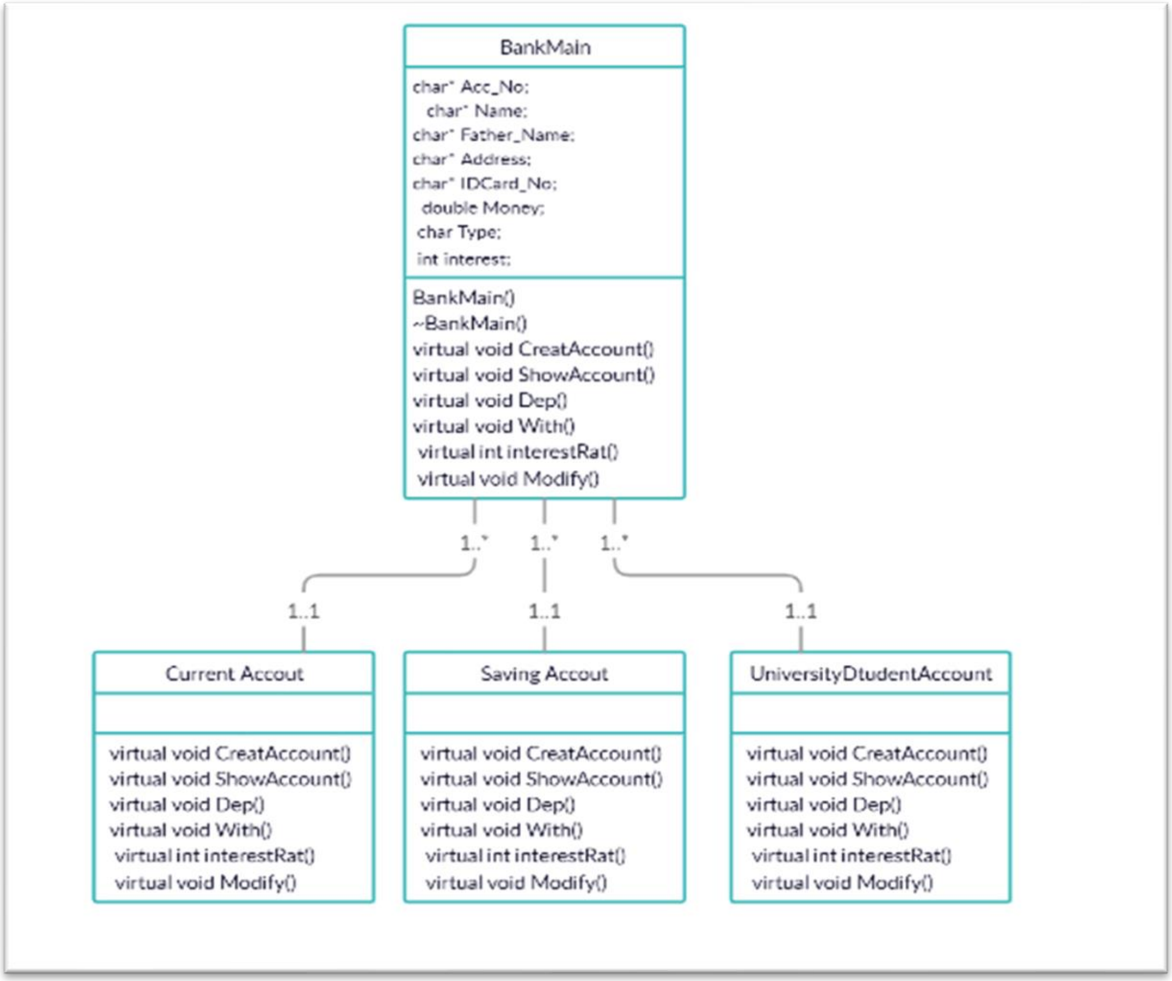
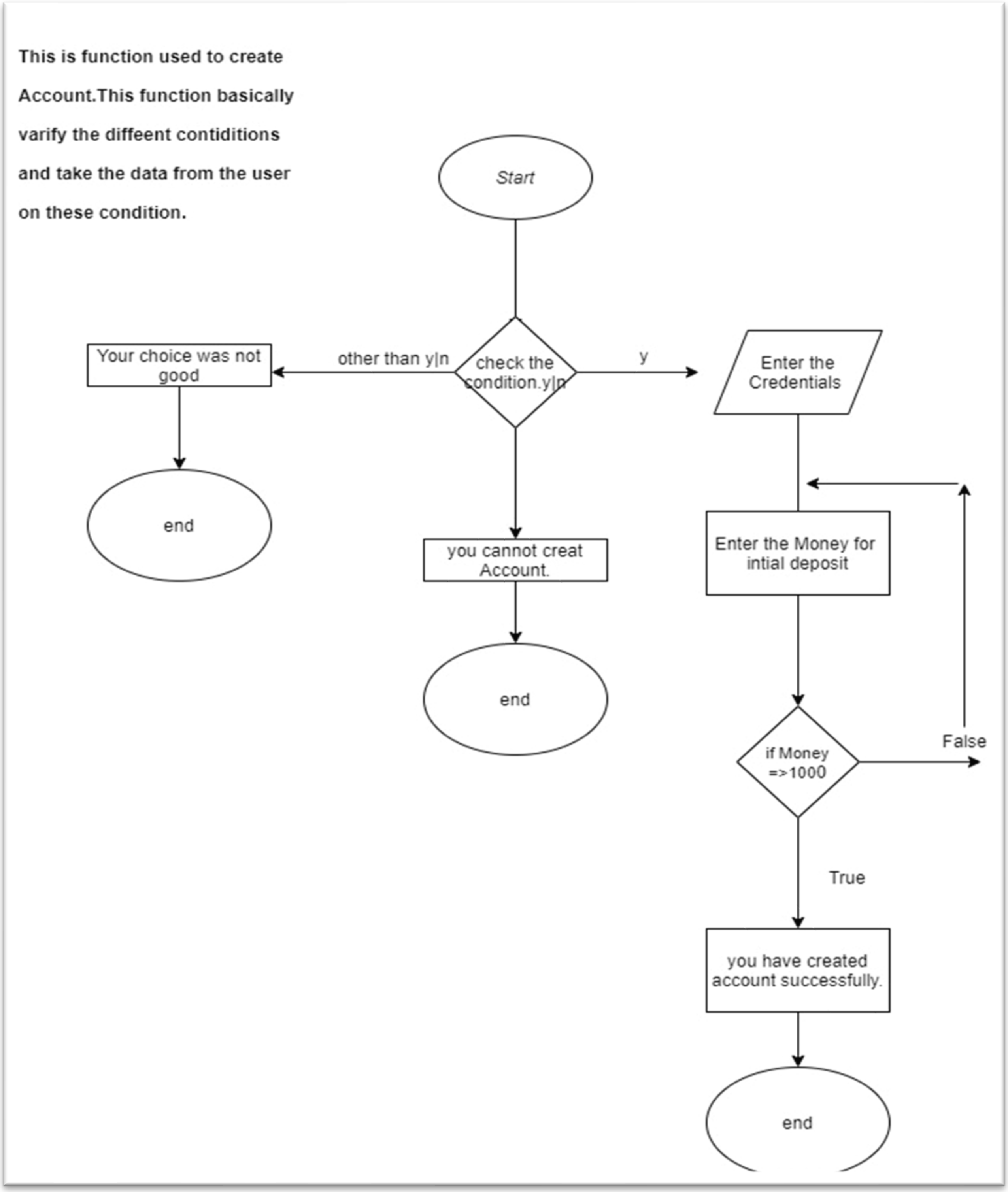


FIGURE 1 :MAIN CLASS INHERITANCE

- ## Functions for Classes:

CreatAccount:



This is function used to create Account.This function basically varify the diffeent contiditions and take the data from the user on these condition.

FIGURE 2:CREAT ACCOUNT

```
write_account():
```

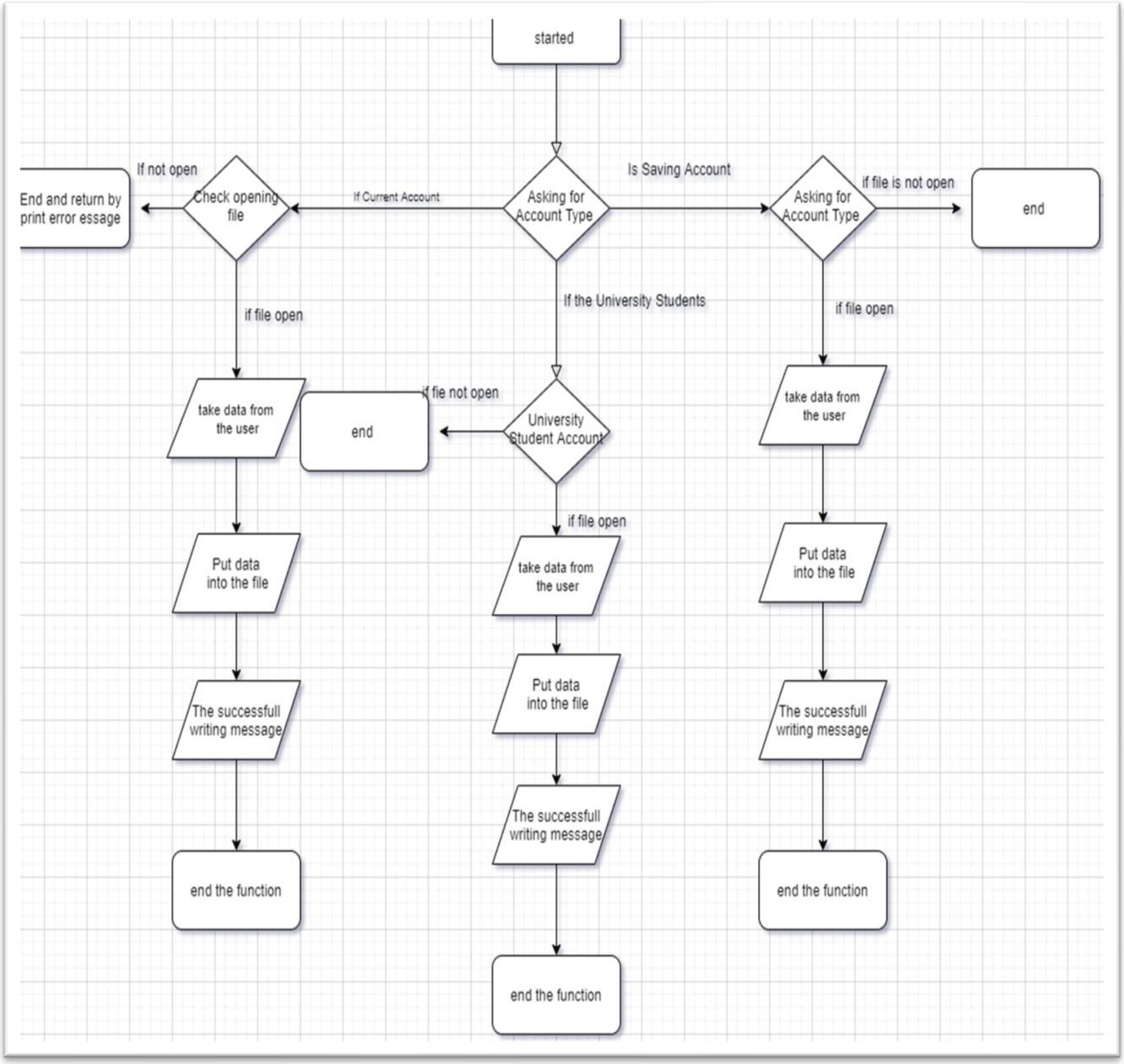


FIGURE 3:WRITE ACCOUNT

`display_all():`

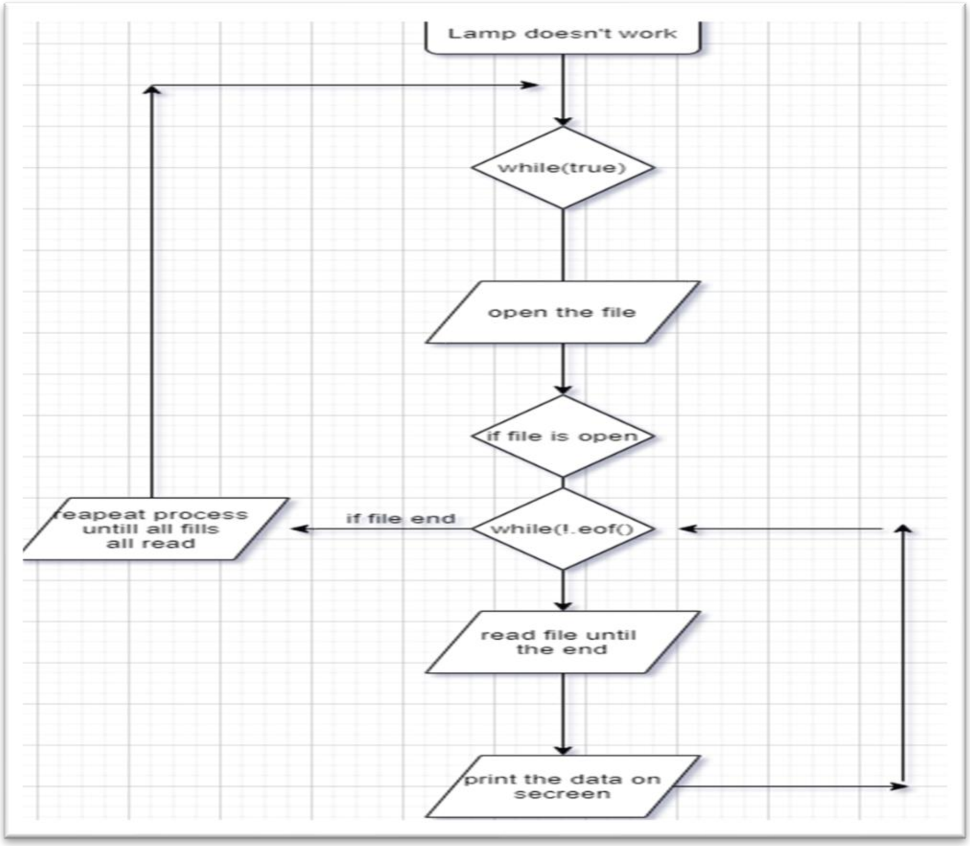


FIGURE 4:DISPLAY ALL THE ACCOUNTS

`Char Selection():`

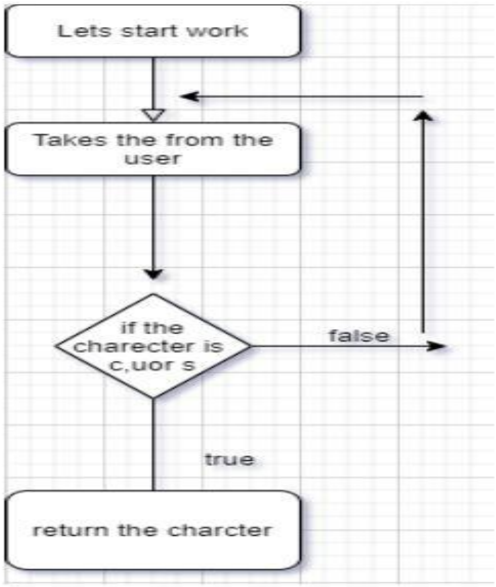


FIGURE 5:SELECT ACCOUN

DISPLAY_SP():

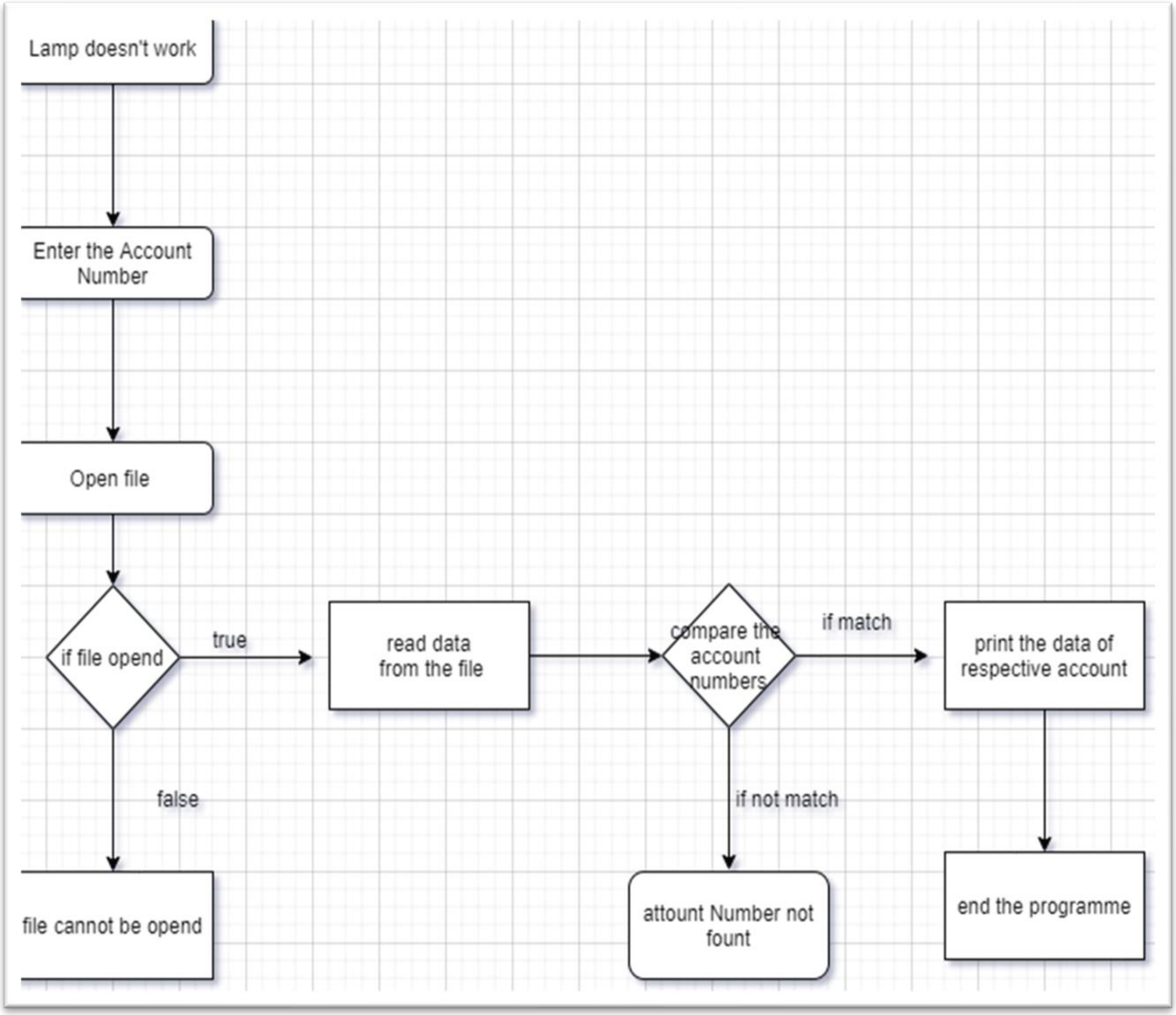


FIGURE 6:DISPLAY A PARTICULAR ACCOUNT

delete_account():

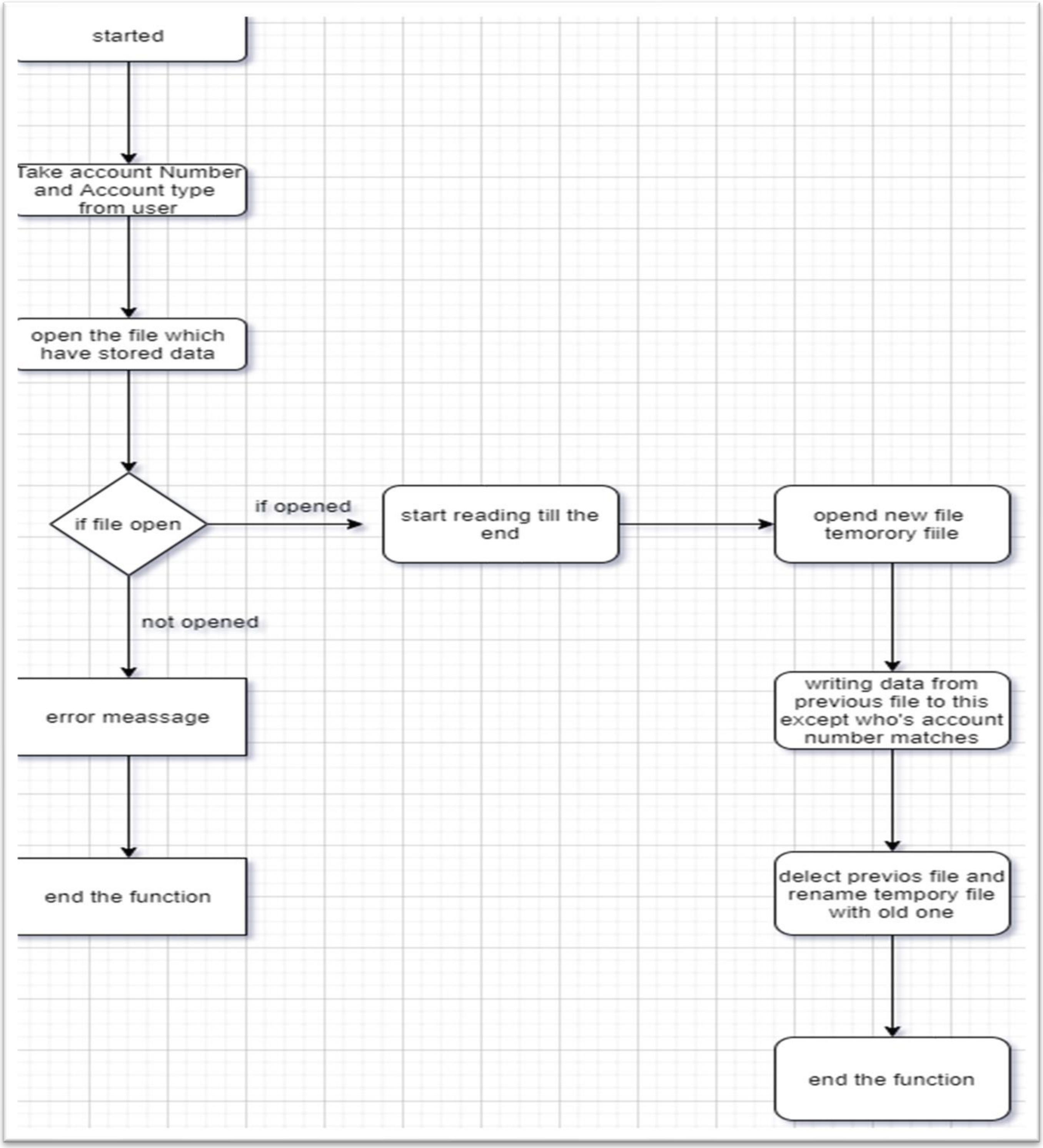

started
Take account Number and Account type from user
open the file which have stored data
if file open
if opened
start reading till the end
opend new file temorory fiile
not opened
writing data from previous file to this except who's account number matches
error meassage
delect previos file and rename tempory file with old one
end the function
end the function

FIGURE 7:DELEET ACCOUNT

modify_account():

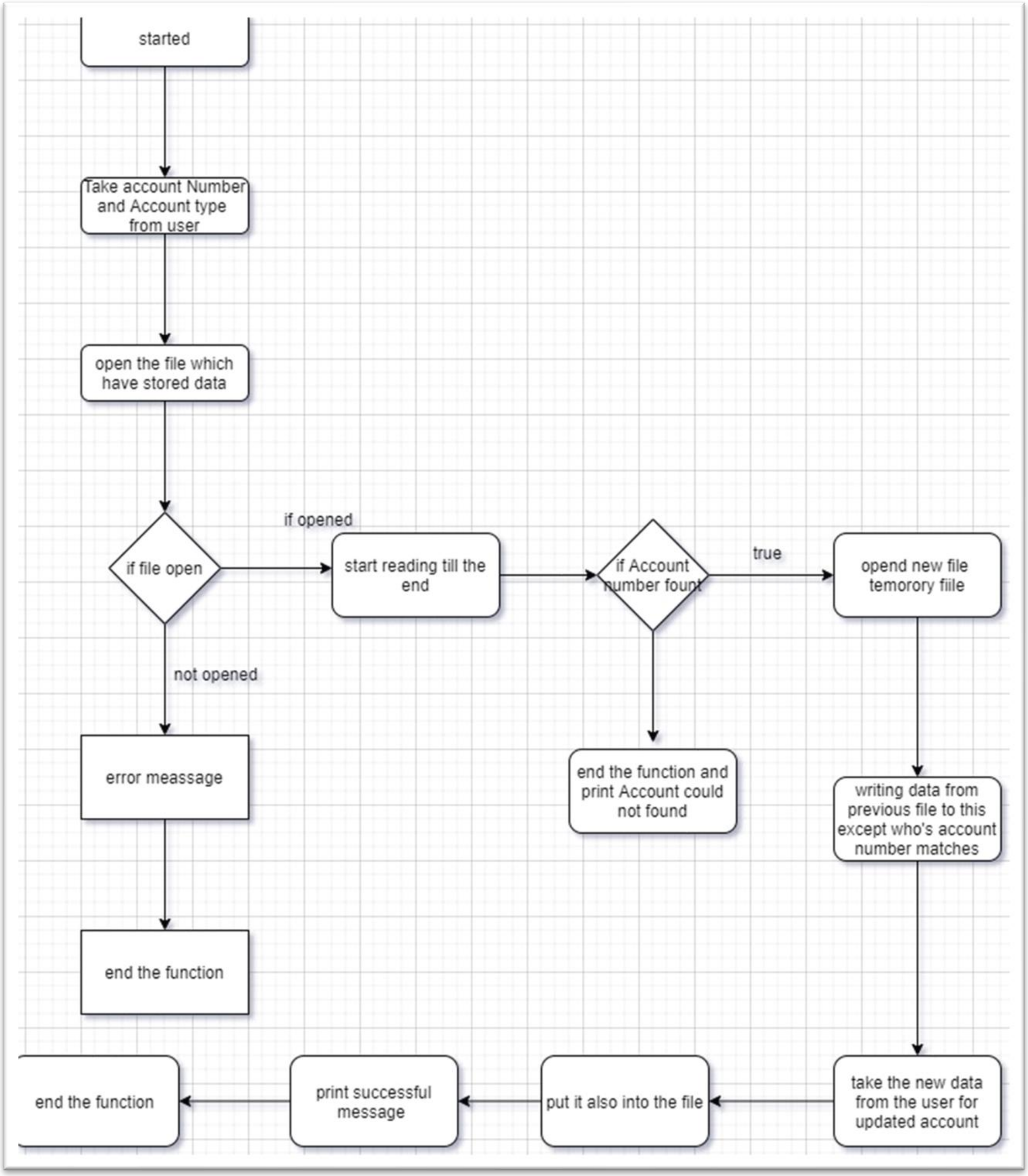

started
Take account Number and Account type from user
open the file which have stored data
if file open
if opened
start reading till the end
if Account number fount
true
opend new file temorory fiile
not opened
error meassage
end the function and print Account could not found
writing data from previous file to this except who's account number matches
end the function
end the function
print successful message
put it also into the file
take the new data from the user for updated account

FIGURE 8:MODIFY ACCOUNTS

ReturnMoney():

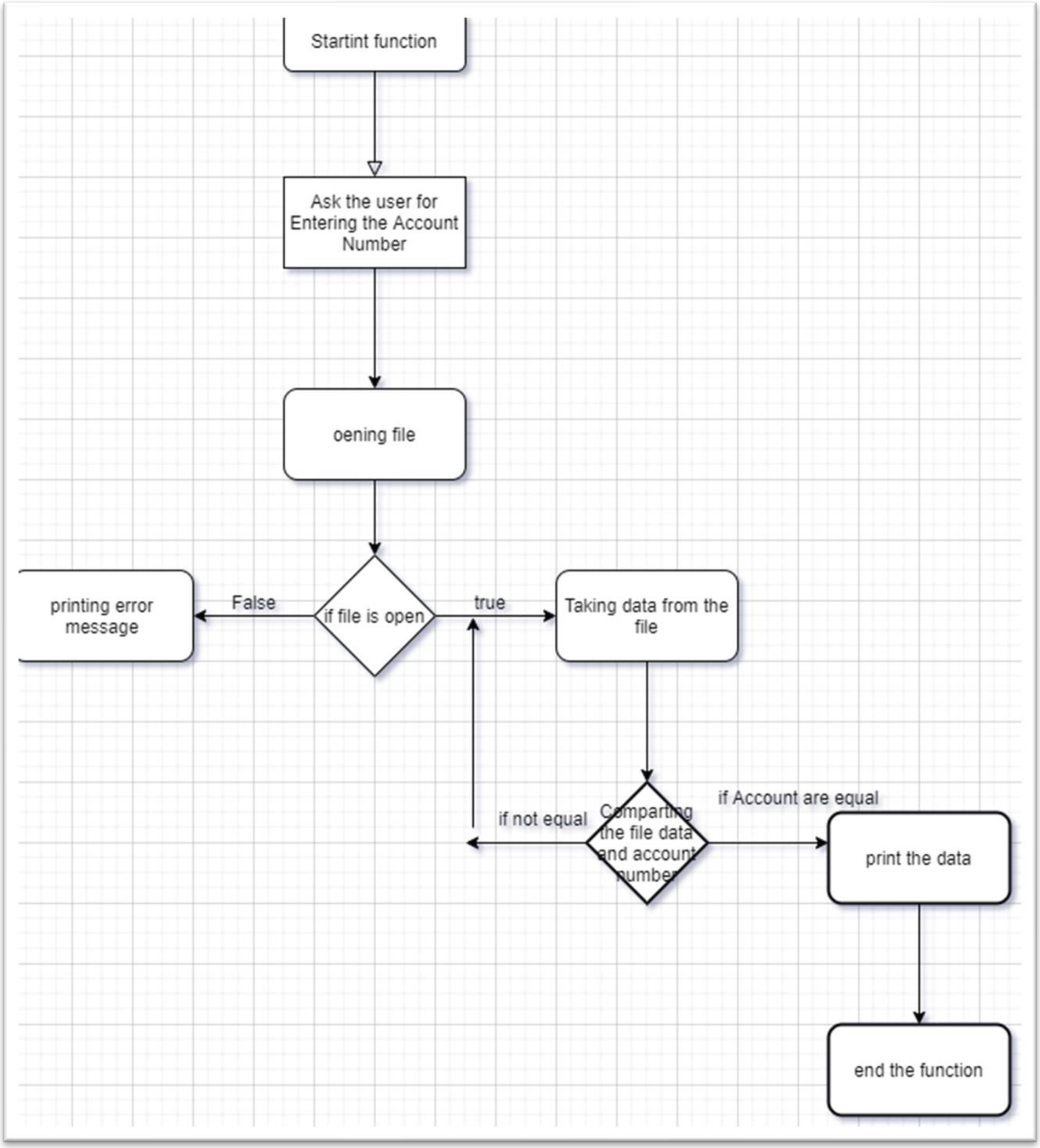


FIGURE 9:RETURN MONEY

3 Program code/Solution:

Now I will explain every bit of code.

Main Class:

First of All, I made the one parent class in which I define all the variables that was common in all the classes. I defined character type pointer for the account Number, Holder's name, Holder's father name, his /her home address, ID card and float type money and character type for type of account. Then define the constructor and destructor and the pure virtual function where I used the polymorphism for the child classes. I defined the constructor and allocated dynamically memory. Then I define constructor where I allocated dynamical memory for the variables. Then this memory is deallocated in the destructor.

Then I define three classes that was child classes of main class. So, all the variables and functions that was defined in the main class now also a part of child classes.

3.1 Current Account Class:

First of all, I defined a child class for current account. Then I defined with all the signatures of functions. This current account has constructor which assigns the account type and interest rate per transaction which is 10.

Then I defined these functions in the other file. First function was there is for creating account. In this function first of all there are certain eligibilities which client has to be fulfill for opening the account like in my home country.

The person who want to open the account must the 18 years of age. He must have the ID card 💳 etc. If he doesn't have these required eligibilities, he cannot open the account. After the verifications we will add the information from the users. Like account number, Name, his father's name, home address and then user can add the initial amount of money at the account opening time.

The amount of money that we enter will differ from the account to account for the Current account user have to enter the amount greater than or equal to the 1000. If the user cannot submit this amount at the opening time then function will ask again and again until he enter the current amount of Money. If the enter the Money greater than or equal to 1000 then a message of successful account opening will appear and function will end. This return true in the case of successful operation and false otherwise.

The I define the function in the current account class showing the account. In this function call it is simply print out all the data on the console screen. Then there is a function for current account class that modify the given data when it is called. It is simply asking for the value of variable to enter and then then assign to the attributes. It has the same condition on the money as the opening time of current account.

3.2 Saving Account Class:

I defined a child class for saving account. Then I define it in header file with all the signatures of functions. This saving account has constructor which assigns the account type and interest rate per transaction which is 10.

Then I defined these functions in the other file. First function was there is for creating account. In this function first of all there are certain eligibilities which client has to be fulfill for opening the account like in my home country.

The person who want to open the account must the 18 years of age. He must have the ID card etc. If he doesn't have these required eligibilities, he cannot open the account. After the verifications we will add the information from the users. Like account number, Name, his father's name, home address and then user can add the initial amount of money at the account opening time.

The amount of money that we enter will differ from the account to account for the Current account user have to enter the amount greater than or equal to the 5000. If the user cannot submit this amount at the opening time then function will ask again and again until he enter the current amount of Money. If the enter the Money greater than or equal to 5000 then a message of successful account opening will appear and function will end. This return true in the case of successful operation and false otherwise.

The I define the function in the current account class showing the account. In this function call it is simply print out all the data on the console screen. Then there is a function for current account class that modify the given data when it is called. It is simply asking for the value of variable to enter and then then assign to the attributes. It has the same condition on the money as the opening time of current account.

3.3 University Student Account Class:

I defined a child class for University Student account. Then I define it in header file with all the signatures of functions. This university account has constructor which assigns the account type and there is no interest rate per transaction in this type of account.

Then I defined these functions in the other file. In this function first of all there are certain eligibilities which client has to be fulfill for opening the account like in my home country.

The person who want to open the account must the 18 years of age. He must have the ID card etc. If he doesn't have these required eligibilities, he cannot open the account. After the verifications we will add the information from the users. Like account number, Name, his father's name, home address there is no restriction in the case for initial deposit money. Then others things work the similar.

3.4 Functions Out of the Classes:

char
Selection (): This is the function that select the account type. It just simply asks the user to enter the account type between the current, saving or for university student account. If user don't enter these types then I used the try throw blocks to throw the exception and taking the input again.

write_account(): This is function that is called in the main testing file to make the account. First if all it will call the selection function to ask for the account number. It will make the account on the basis of entered type from the user. This function first of all will open the file for writing the data.

If it will open then it will call the create function of respect account class and assign the value to the variables and then write it into the file. And the successful message will appear. If we enter the wrong type then it will ask to enter the correct account type.

void
display_all(): This function is simply read the all data from the all three type of account type files and then print it on the screen. First of all it will open the file were the current account type data is stored after reading it ,it will call the print data function from the class and continues until the end of file as I am using the while loop and going until the end of the file .It will repeat the same process for data reading for other two types of account's files.

display_sp():

This is a function that will ask the user to enter the account type by calling the selection function and account number. After this this function will open the file and started to reading data from the file and comparing its account number to enter by the user .There is also an flag used that is incremented only in the case of account matches and print its all the credentials by using the display function respective class .This process is also repeated for others type of accounts' files. In the case the account could not found it mean the flag would not be incriminated and the message of account could not find will be appear.

modify_account():

In this function users are asking to enter the account number and account type by using the select function. Then file is opened and started to read data from this function. Another file is also open and started to read data from previously opened file to newly opened file except with the data of the account number and account type entered by the user. Then and modify function of the class is called and its data is written into the temporarily open file. Now temporarily opened file has the data of previous file except the account number entered by the user and the newly entered data. Then the previously opened file is deleted and temporarily file is renamed with the previously opened file name. Then process is repeated for every account type if the user enters this type. A flag is also used which is updated only in the case of account matches otherwise it remains same. If it does not update the its means that this account of particular account number and account type doesn't exist.

dalete_account():

This is a function which works similarly like the modifying function except calling the modify function and writing the data of modified function into the file. As a file of the particular account type is opened. User entered the account number that he wants to delete. We open a temporary file in which data is written from the previously opened file to the temporarily opened file except with account number entered by the user. A flag is updated if the account number matches. And them the previously opened file is deleted and temporarily opened file is renamed with the name of previously opened file .And then this process is repeated for other two account's files too. If the flag could not update then its means that the account could not found in this case the message of account could not found is printed.

ReturnMoney():

In this function the user is asked to enter the account number and the flag is defined. Then a file is opened and the started to read data from the data and comparing its account number to the entered by the user. If it matches the money of this account is printed and the flag is updated. This is repeated for every file. Data is read until the end of file. If the flag is not updated its mean that this account doesn't exist.

Interest():

In this function the user is asked to enter the account type and the interest rate function of this account type is called and integer type result is printed.

ExchangeRate():

In this function a file in which I have already have stored the all the exchange rate other currencies to Hungarian forints are printed just reading the file. If it could not open the its mean there is a problem or we could not have this file .Then an error message is printed. It is necessary to add the file exchange before reading it.

ExchangeRateCal():

This function is just calculating the exchanged amount of Money from others account to Hungarian Forint's. First there will their list of available exchange options will be appearing on the screen. User have to enter the option and the enter the amount of Money that he wants to exchange from other account to Hungarian forints.

Main Testing File:

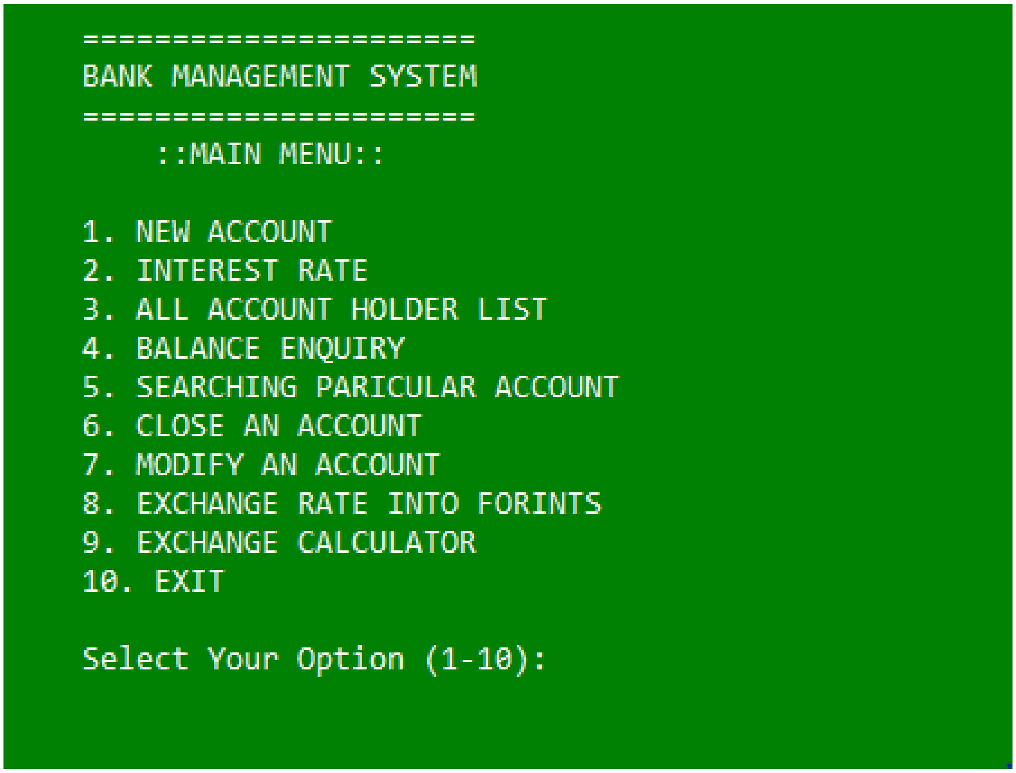
In this main testing file, I have written all the functions prototypes that I have to call for the implementation on the console windows. Then use the switches statements to give the user different options what actually he wants to do. By selecting right options, the user can do any work from the available options.

4 External Resources:

The external resource that I used was to change the color of consoles window and the brightness of appeared txt. The link of this resource is : https://youtu.be/kiqqxuu1gRk

5 Testing:

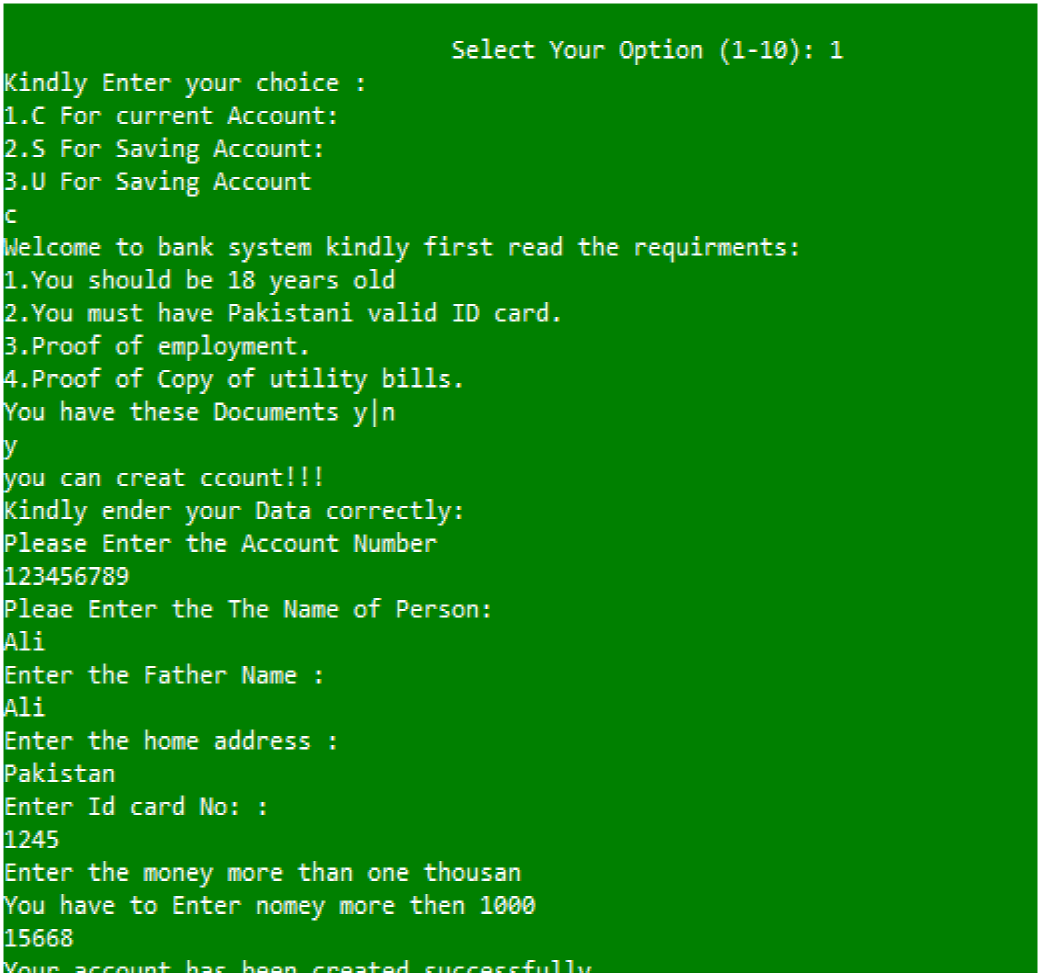
5.1 Console view



```
======================
BANK MANAGEMENT SYSTEM
======================
    ::MAIN MENU::

1. NEW ACCOUNT
2. INTEREST RATE
3. ALL ACCOUNT HOLDER LIST
4. BALANCE ENQUIRY
5. SEARCHING PARICULAR ACCOUNT
6. CLOSE AN ACCOUNT
7. MODIFY AN ACCOUNT
8. EXCHANGE RATE INTO FORINTS
9. EXCHANGE CALCULATOR
10. EXIT

Select Your Option (1-10):
```

FIGURE 10:CONSOLE VIEW

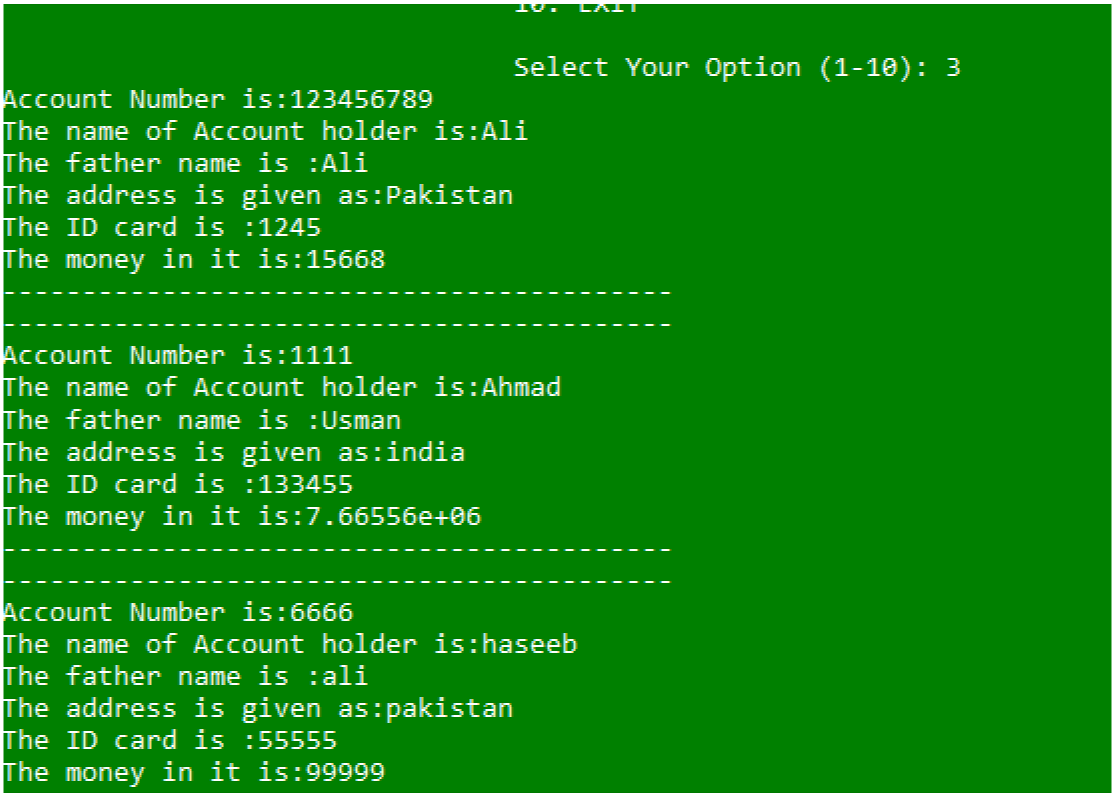
5.2 Account Making



```
                                Select Your Option (1-10): 1
Kindly Enter your choice :
1.C For current Account:
2.S For Saving Account:
3.U For Saving Account
c
Welcome to bank system kindly first read the requirments:
1.You should be 18 years old
2.You must have Pakistani valid ID card.
3.Proof of employment.
4.Proof of Copy of utility bills.
You have these Documents y|n
y
you can creat ccount!!!
Kindly ender your Data correctly:
Please Enter the Account Number
123456789
Pleae Enter the The Name of Person:
Ali
Enter the Father Name :
Ali
Enter the home address :
Pakistan
Enter Id card No: :
1245
Enter the money more than one thousan
You have to Enter nomey more then 1000
15668
Your account has been created successfully
```

Figure 11:Account Making

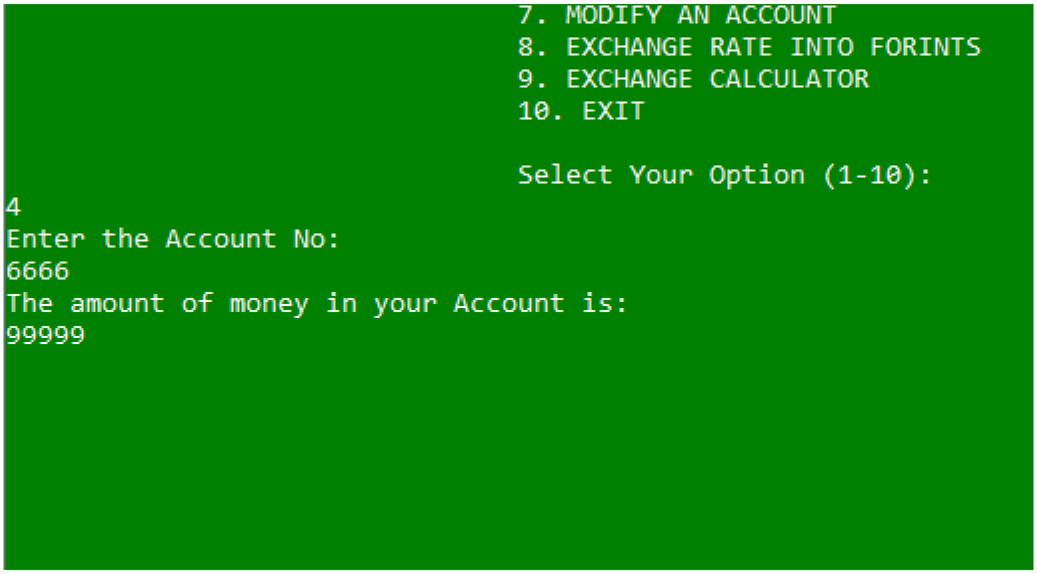
5.3 Print All the Account Holder’s



```
                                10. EXIT

                                Select Your Option (1-10): 3
Account Number is:123456789
The name of Account holder is:Ali
The father name is :Ali
The address is given as:Pakistan
The ID card is :1245
The money in it is:15668
------------------------------------------
------------------------------------------
Account Number is:1111
The name of Account holder is:Ahmad
The father name is :Usman
The address is given as:india
The ID card is :133455
The money in it is:7.66556e+06
------------------------------------------
------------------------------------------
Account Number is:6666
The name of Account holder is:haseeb
The father name is :ali
The address is given as:pakistan
The ID card is :55555
The money in it is:99999
```

FIGURE 12:PRINT ALL THE ACCOUNT HOLDER’S

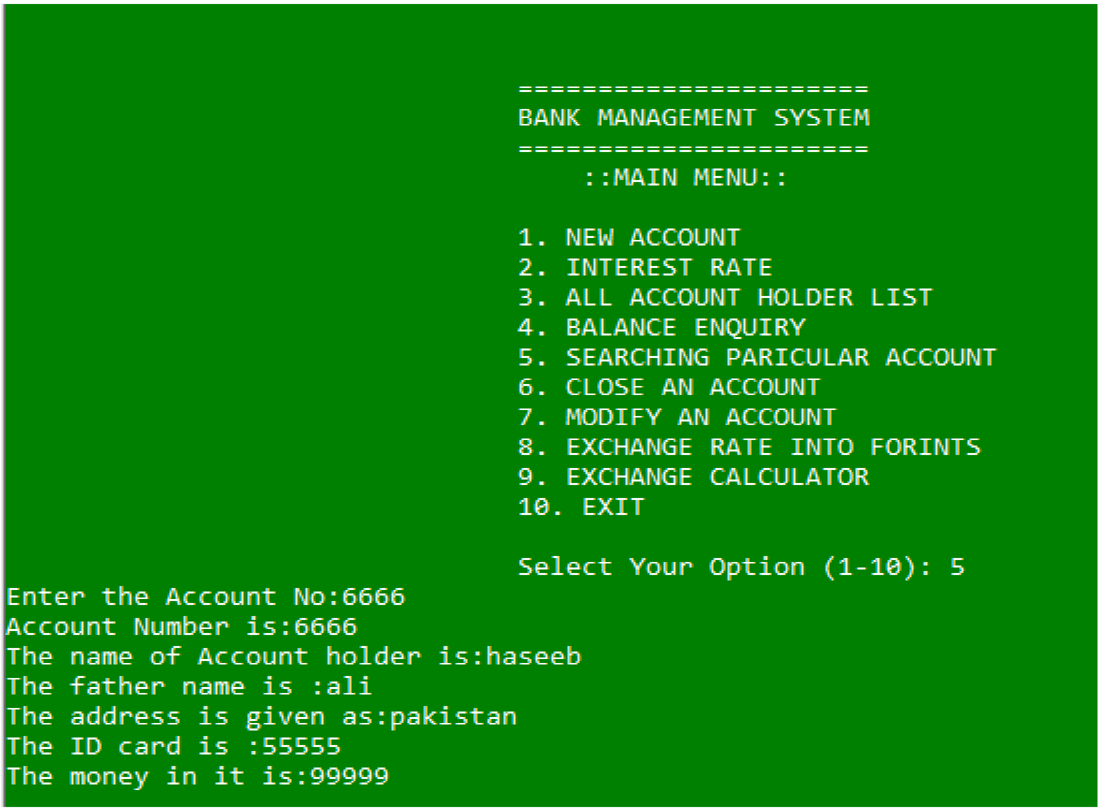
5.4 Balance Enquiry



```
                                7. MODIFY AN ACCOUNT
                                8. EXCHANGE RATE INTO FORINTS
                                9. EXCHANGE CALCULATOR
                                10. EXIT

                                Select Your Option (1-10):
4
Enter the Account No:
6666
The amount of money in your Account is:
99999
```

FIGURE 13:BALANCE ENQUIRY

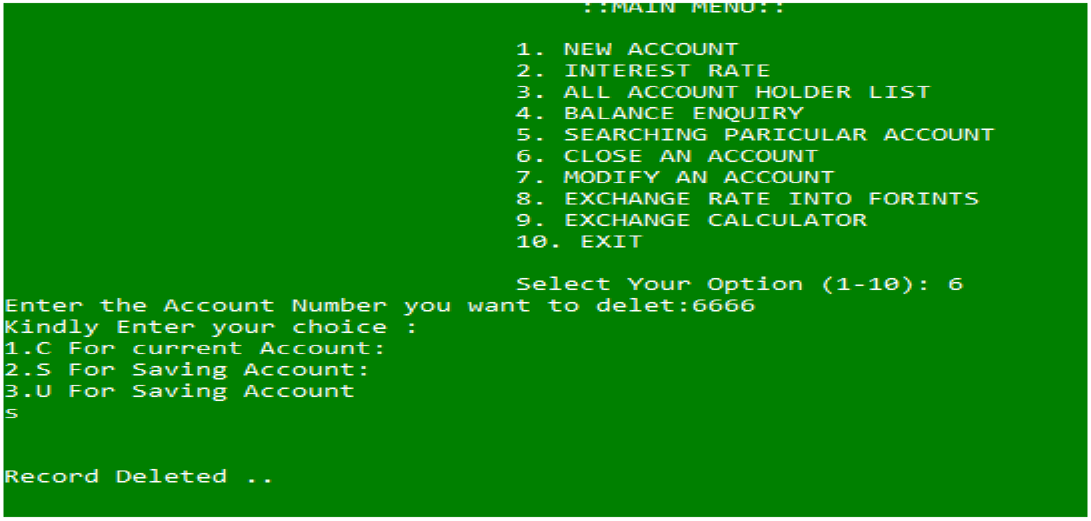
5.5 Search An Account



```
                                ======================
                                BANK MANAGEMENT SYSTEM
                                ======================
                                    ::MAIN MENU::

                                1. NEW ACCOUNT
                                2. INTEREST RATE
                                3. ALL ACCOUNT HOLDER LIST
                                4. BALANCE ENQUIRY
                                5. SEARCHING PARICULAR ACCOUNT
                                6. CLOSE AN ACCOUNT
                                7. MODIFY AN ACCOUNT
                                8. EXCHANGE RATE INTO FORINTS
                                9. EXCHANGE CALCULATOR
                                10. EXIT

                                Select Your Option (1-10): 5
Enter the Account No:6666
Account Number is:6666
The name of Account holder is:haseeb
The father name is :ali
The address is given as:pakistan
The ID card is :55555
The money in it is:99999
```

FIGURE 14:SEARCH AN ACCOUNT

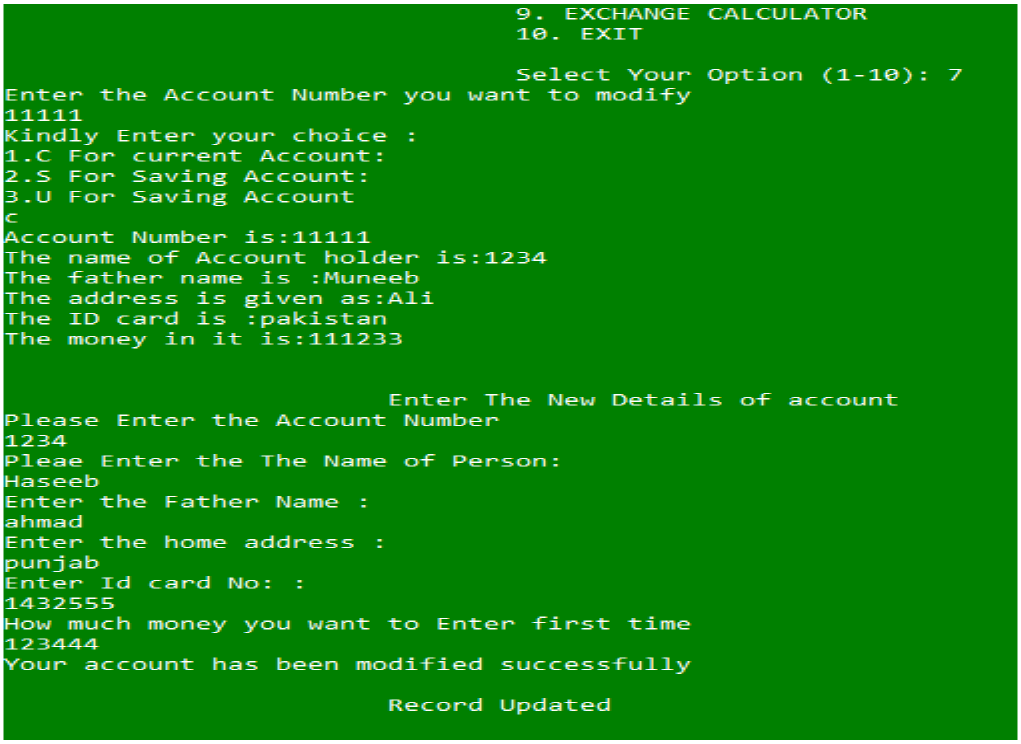
5.6 Closing an account



```
                                    ::MAIN MENU::

                                1. NEW ACCOUNT
                                2. INTEREST RATE
                                3. ALL ACCOUNT HOLDER LIST
                                4. BALANCE ENQUIRY
                                5. SEARCHING PARICULAR ACCOUNT
                                6. CLOSE AN ACCOUNT
                                7. MODIFY AN ACCOUNT
                                8. EXCHANGE RATE INTO FORINTS
                                9. EXCHANGE CALCULATOR
                                10. EXIT

                                Select Your Option (1-10): 6
Enter the Account Number you want to delet:6666
Kindly Enter your choice :
1.C For current Account:
2.S For Saving Account:
3.U For Saving Account
s


Record Deleted ..
```

FIGURE 15:CLOSING AN ACCOUNT

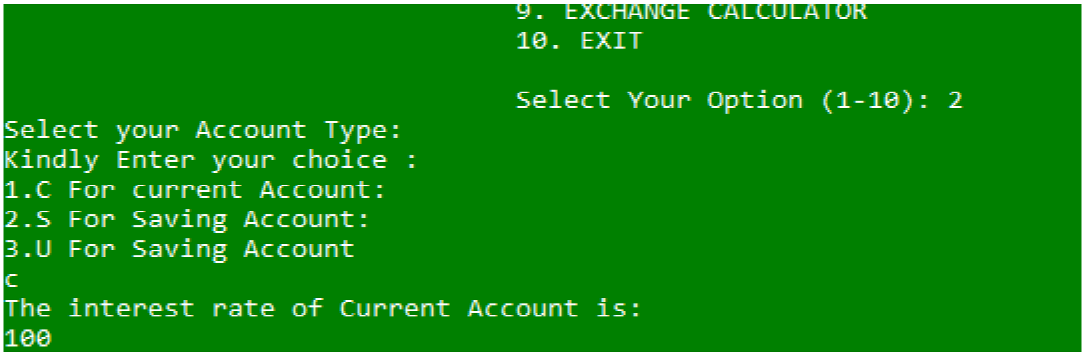
5.7 Modifying and Account



```
                                9. EXCHANGE CALCULATOR
                                10. EXIT

                                Select Your Option (1-10): 7
Enter the Account Number you want to modify
11111
Kindly Enter your choice :
1.C For current Account:
2.S For Saving Account:
3.U For Saving Account
c
Account Number is:11111
The name of Account holder is:1234
The father name is :Muneeb
The address is given as:Ali
The ID card is :pakistan
The money in it is:111233


                        Enter The New Details of account
Please Enter the Account Number
1234
Pleae Enter the The Name of Person:
Haseeb
Enter the Father Name :
ahmad
Enter the home address :
punjab
Enter Id card No: :
1432555
How much money you want to Enter first time
123444
Your account has been modified successfully

                        Record Updated
```

FIGURE 16:MODIFYING AND ACCOUNT

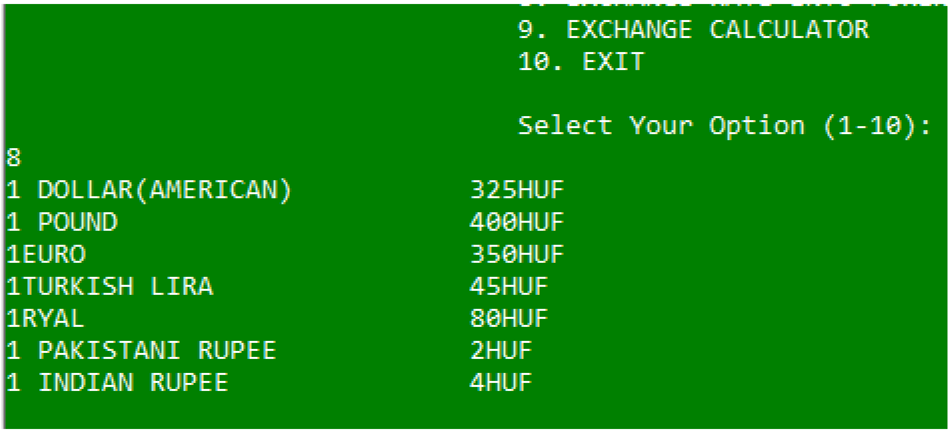
5.8 Interest Rate



```
                                9. EXCHANGE CALCULATOR
                                10. EXIT

                                Select Your Option (1-10): 2
Select your Account Type:
Kindly Enter your choice :
1.C For current Account:
2.S For Saving Account:
3.U For Saving Account
c
The interest rate of Current Account is:
100
```

FIGURE 17:5INTEREST RATE

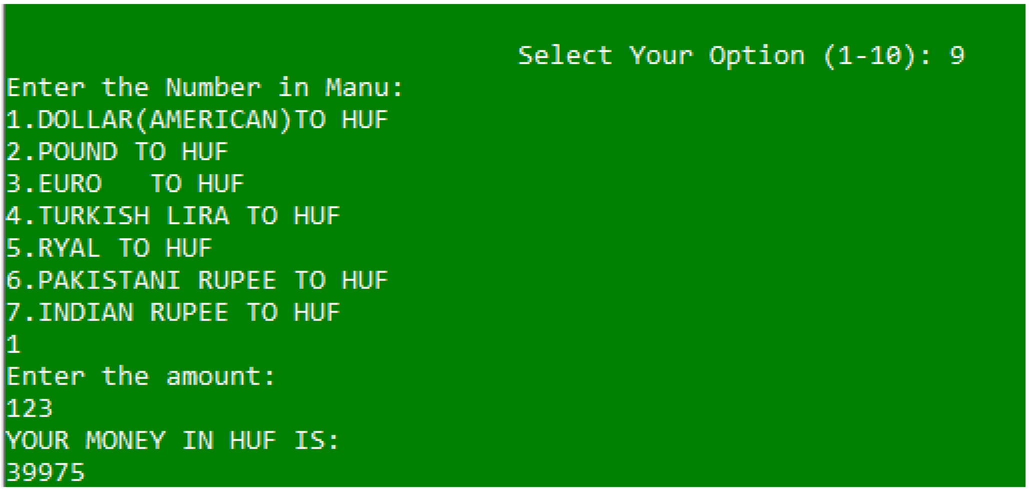
5.9 Today's Exchange Rate



```
                                9. EXCHANGE CALCULATOR
                                10. EXIT

                                Select Your Option (1-10):
8
1 DOLLAR(AMERICAN)             325HUF
1 POUND                        400HUF
1EURO                          350HUF
1TURKISH LIRA                  45HUF
1RYAL                          80HUF
1 PAKISTANI RUPEE              2HUF
1 INDIAN RUPEE                 4HUF
```

Figure 18:Today's Exchange Rate

5.10Exchange Calculator



```
                                Select Your Option (1-10): 9
Enter the Number in Manu:
1.DOLLAR(AMERICAN)TO HUF
2.POUND TO HUF
3.EURO   TO HUF
4.TURKISH LIRA TO HUF
5.RYAL TO HUF
6.PAKISTANI RUPEE TO HUF
7.INDIAN RUPEE TO HUF
1
Enter the amount:
123
YOUR MONEY IN HUF IS:
39975
```

Figure 19:Exchange Calculator

6 What is Different from original:

I tried best as I can to make the perfect project. I something that was different from my original planned project like I added two features like the I added the calculator for the currency exchange and printing the currency exchange rate. I also added the interest rate for different amount of accounts. I excluded the money deposit and withdraw portion because it was the part of project like ATM which I think out of the scope of project and the project's length was increasing.

7 Technique used:

7.1 Necessary Techniques

- an object-oriented approach to programming,
- File Handling
- Exception handling
- Dynamic memory allocation

7.2 Extra Technique

- Inheritance
- Polymorphism:

8 Project Summary:

This was my project in which I used the all the necessary features like object-oriented approach to programming, dynamic memory management, exception handling, file management and some other things like the Inheritance (base and derived classes) and Polymorphism: abstract classes and virtual functions. I also added the code complexity, readability and commenting to be understood easily.

I take the tips for documentation and commenting via online resource.

9 Reference

https://youtu.be/kiqqxuu1gR

http://www.edparrish.net/common/cppdoc.html