 **RNS Institute of Technology, Bengaluru – 98**

(AICTE Approved, VTU Affiliated and NAAC 'A' Accredited)

**Department of Information Science and Engineering**

(Accredited by NBA for the Academic years 2018-19, 2019-20 and 2020-2021)

**Data structures Laboratory-18CSL38**

**Mini Project Evaluation**

|  |  |  |  |
| --- | --- | --- | --- |
| **Project Title** | BANKING SYSTEM | | |
| **USN** | 1RN19IS122 | 1RN19IS123 | 1RN19IS124 |
| **Name** | ROHITH KUMAR HK | RUSWANTH.K | RUTVICK SREEDHAR |

**Abstract**

The purpose of this mini project is in fulfillment of the requirements of customer using the bank for payment. The programing language used to develop the project is C. The data structure used is doubly linked list.

The domain “Banking System” keeps the day to day Tally record as a complete Banking. It can keep the information of Account opening, Deposit, Withdraw and display the transaction report, Individual account opening and editing details. The exciting part of this project is it displays transaction reports, Statistical summary of all accounts.

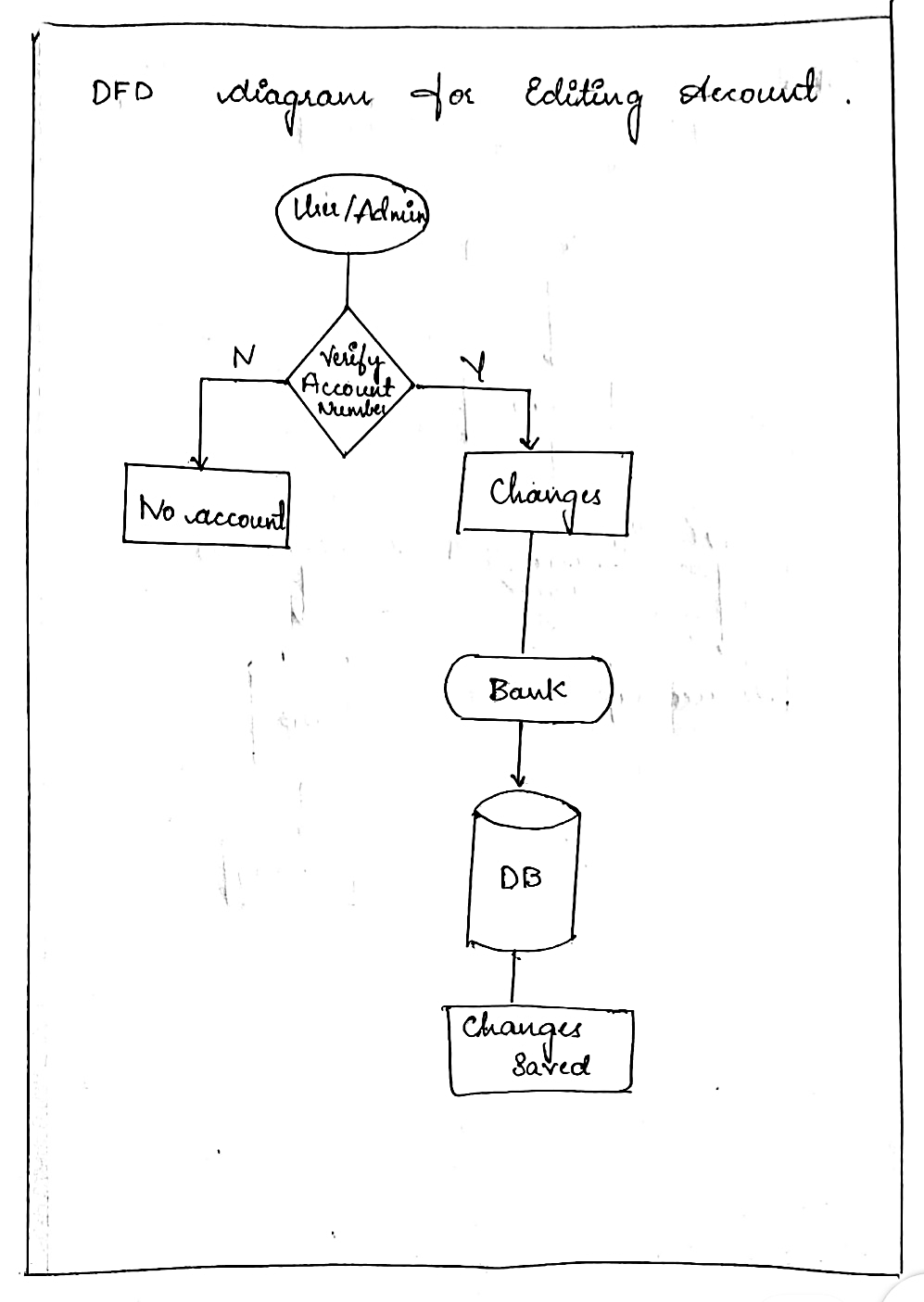
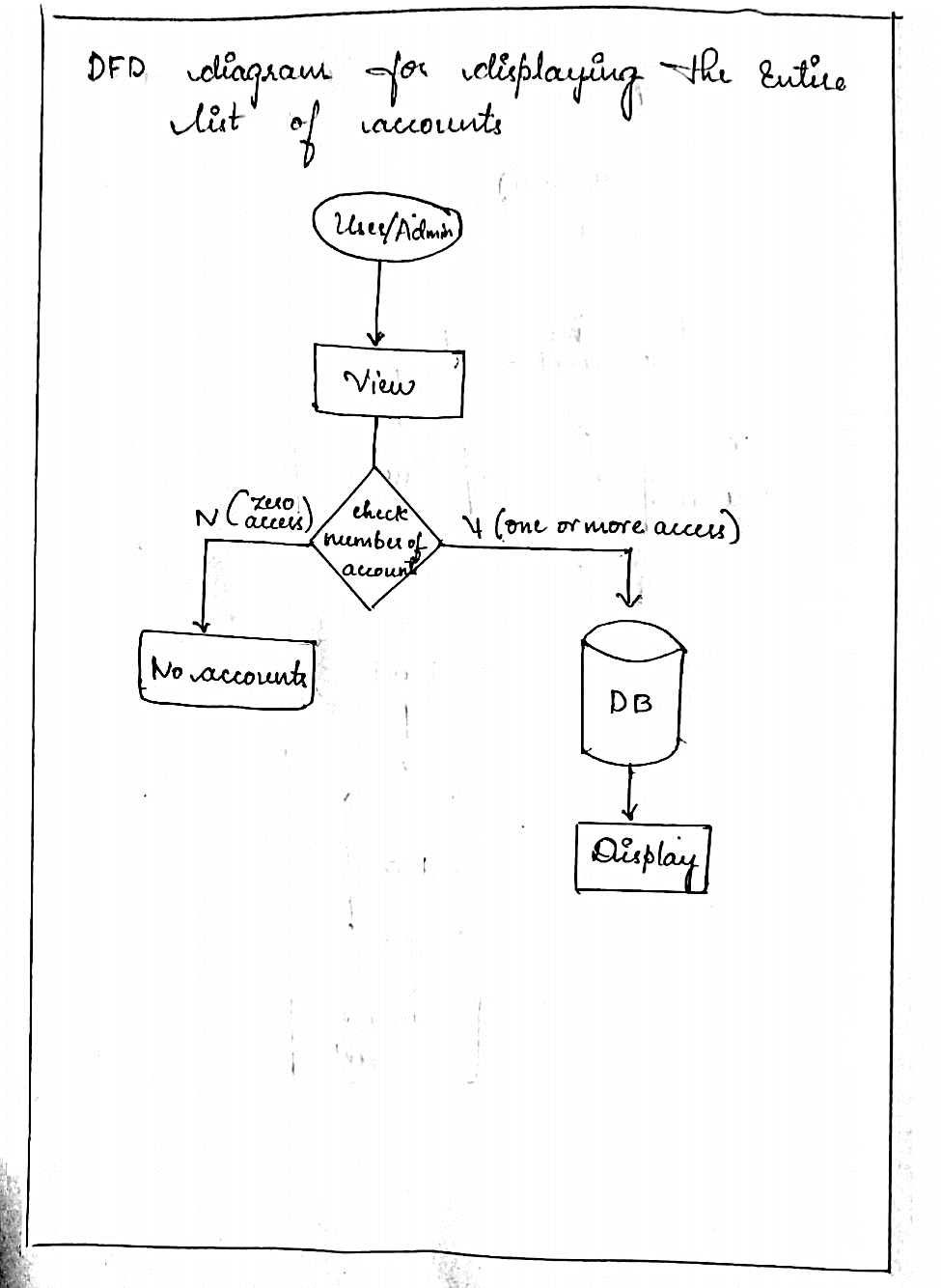
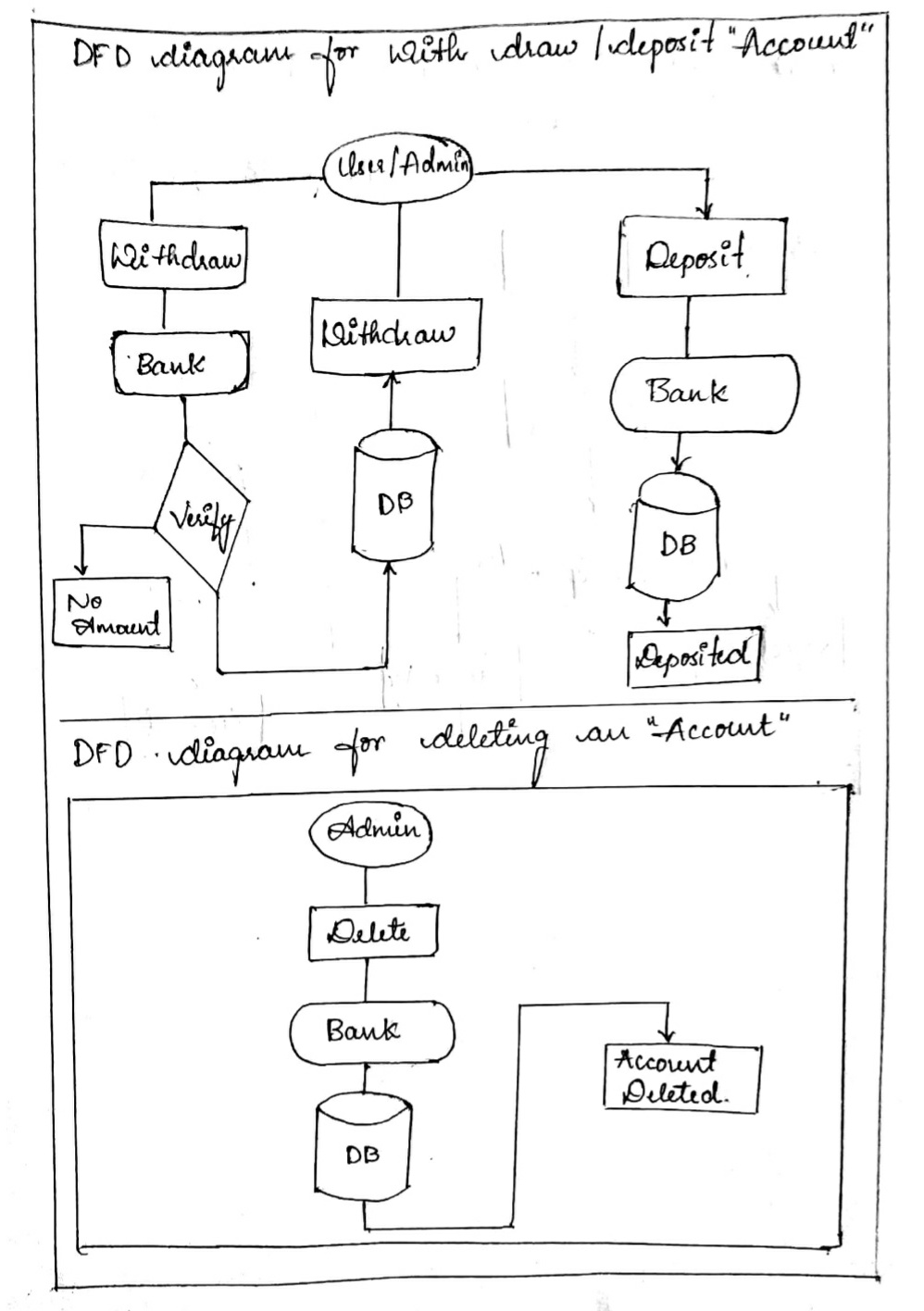
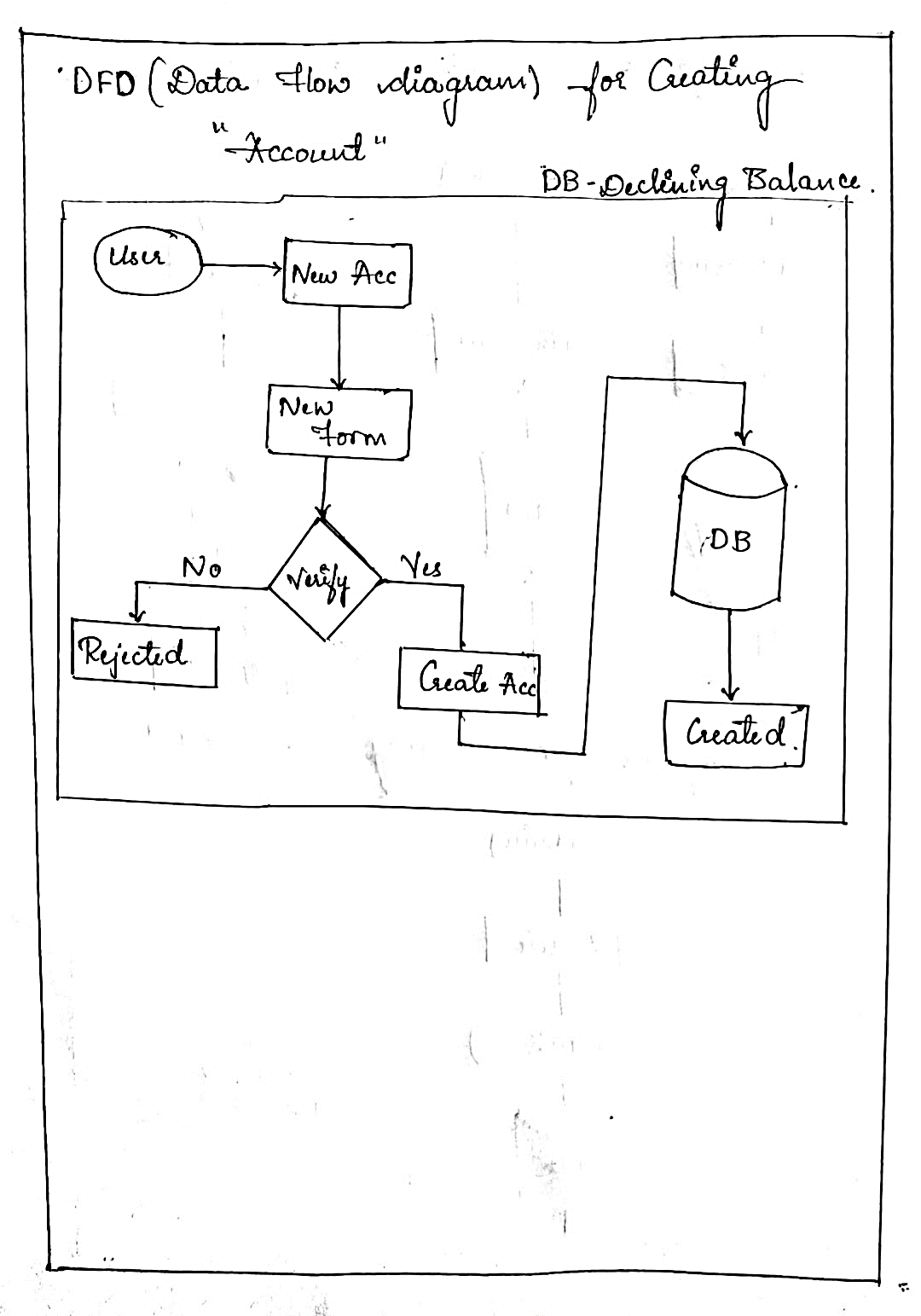
**Introduction**

This program will be provided as toll to the bank. The bank has been working for account information, withdrawal (through cash/cheque) and deposit amount. In this program you can keep record for daily banking transactions.

The objective is to prepare the program that can maintain data and provide a user friendly interface for retrieving customer related details just in few seconds, so it is not time consuming process. No paper work required and can be implemented further.

The program should also facilitate the addition of new customer account , deletion of account and modification of existing customer account and show all.

**Description about the project work**

**

**Node (structure)**

**Variables declared:**

**Char types:**

Fname(first name)

Lname(last name)

Acc\_no(account number)

Address

Citizenship (aadhar number)

Phone(mobile number)

**Int** **types:**

Age

**Float types:**

Amt(amount/balance)

**Pointer variables**:

First(head)

Last(tail)

Temp(pointer1)

Temp1(pointer2)

Prev

Next

**Functions:**

1] **create()**

Step1: input account number (8digits).

Step2: verify whether it’s 8 digits and no alphabets in between and store it.

Step3: input first name and last name, age, address and store it.

Step4: input aadhar number (12 digits).

Step5: verify whether its 12digits and no alphabets and store it.

Step6: input phone number (10 digits).

Step7: verify whether it’s 10 digits and no alphabets and store it.

Step8: input amount to deposit.

Step9: return.

2] **insertatfirst()**

Step1: calls create() function and input values.

Step2: if first == NULL

Then

first=temp

last=first

else

first-> prev= temp

Temp->next=first

First=temp

Step4: return.

3] **display()**

Step1: if first == NULL

Then

Print no accounts.

Else

While(temp!= NULL)

Print the list of accounts.

temp= temp->next

End while.

Step2: return.

4] **edit()**

Step1: input account number to edit.

Step2: verify account number for existence.

True-> goto step3 to 6.

False-> print ‘no accounts’ return.

Step3: print options { 1. to edit name

2. To edit age

3. To edit address

4. To edit phone number

5. To edit aadhar number}

Step4: input choice.

Step5: input values to be changed.

Step6: verify input then value are changed and saved.

Step7: return.

5] **depositamt()**

Step1: input account number

Step2: verify account number

If it exits then

Goto step3

Else print ‘account number does not exist’

Return.

Step3: print ‘account details and balance’.

Step4: input amount to deposit.

Step5: balance updated.

Step6: print ‘account details and new balance’.

Step7: return.

6] **withdrawamt()**

Step1: input account number.

Step2: verify account number

If it exits then

Goto step3

Else

Print ‘account number does not exit’

Return.

Step3: print ‘account details and balance’.

Step4: input amount to withdraw

Step5: if amount > balance

Print ‘insufficient balance …. Retry’

Goto step4

Else

Balance = balance – withdraw

Step6: print account details and new balance.

Step7: return.

7] **delaccfront()**

Step1: if first == NULL

Print ‘no account to delete’.

Return.

Step2: if temp-> next == NULL

Print account details that is deleted.

Free(temp)

First= null

Else

First=temp->next

Print account details that is deleted.

Free(temp)

First->prev= NULL.

Step3: return.

8] **delaccend()**

Step1: if first == NULL

Print ‘no of account to delete’

Return.

Step2: if temp->next == NULL

Print account details that is deleted.

Free (last)

Last= temp1

Last-> next = NULL

Step3: return

9] **delacc()**

Step1: if first == NULL

Print ‘no of account is delete’

Return

Step2: if temp-> next == NULL

Print account details to be deleted

Free (temp)

First = NULL

Return.

Step3: input account number is deleted

Step4: verify account number

True-> goto step5

False-> print ‘account does not exist’

Return.

Step5: while(temp!= NULL)

If (temp ->prev== NULL)

Calls dellaccfront()

Break;

Else

Print account details to be deleted.

Tmp-> prev-> next= temp->next

Temp-> next-> prev= tenp-> prev

Free(temp

Break;

Temp= temp-> next

End while.

Step6: return.

10] **main()**

Step1: print menu

Step2: print choice { 1.create number of new account

2. Add another account

3. View accounts list

4. Edit account

5. Deposit amount

6. Withdraw amount

7. Delete account

8. Exit}

Step3: input choice

Step4: while(1)

Step5: switch(choice)

case 1: input value of n

for i=0 to i<n

call insertatfirst()

i++

end for

break;

case 2: insertatfirst()

break

case 3: display()

break

case 4: edit()

break

case 5: depositamt()

break

case 6: withdrawamt()

break

case 7: delacc()

break

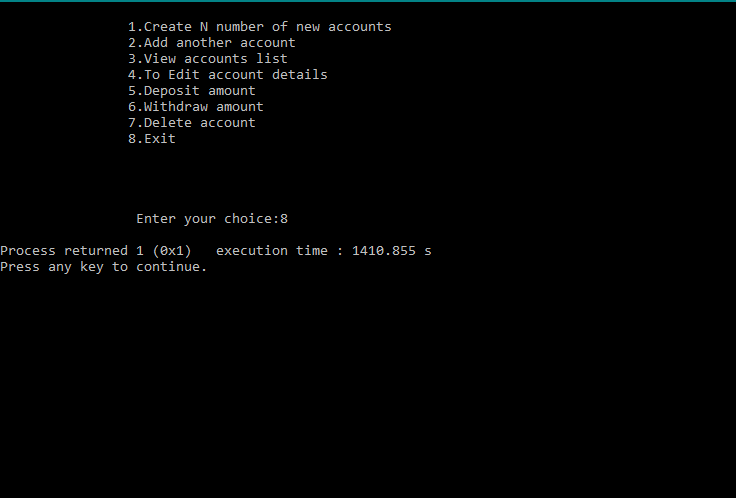
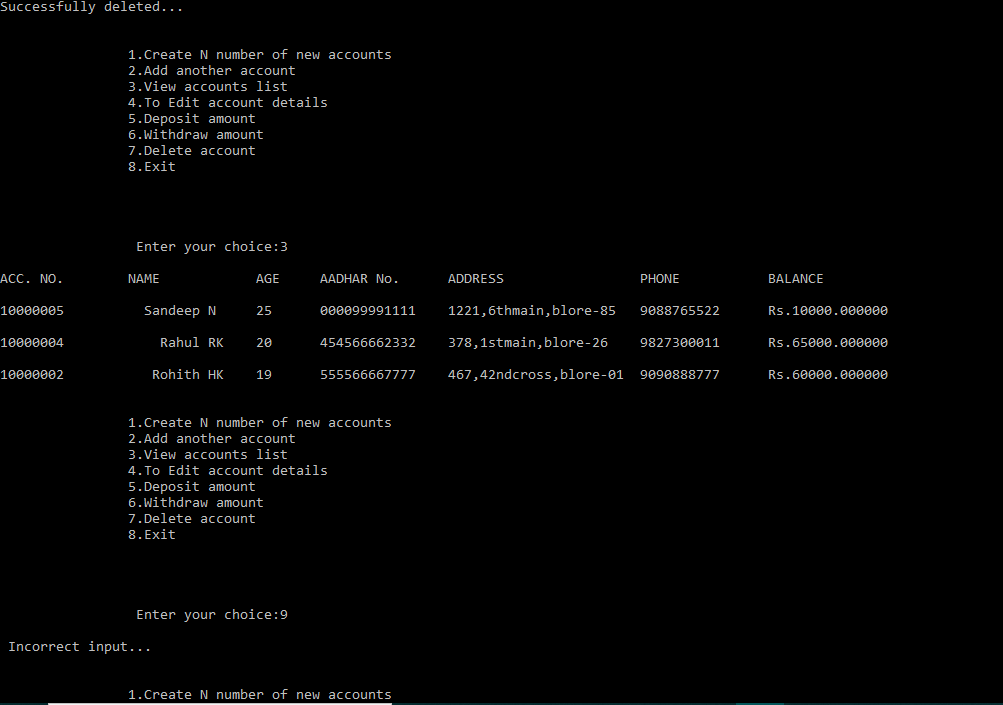
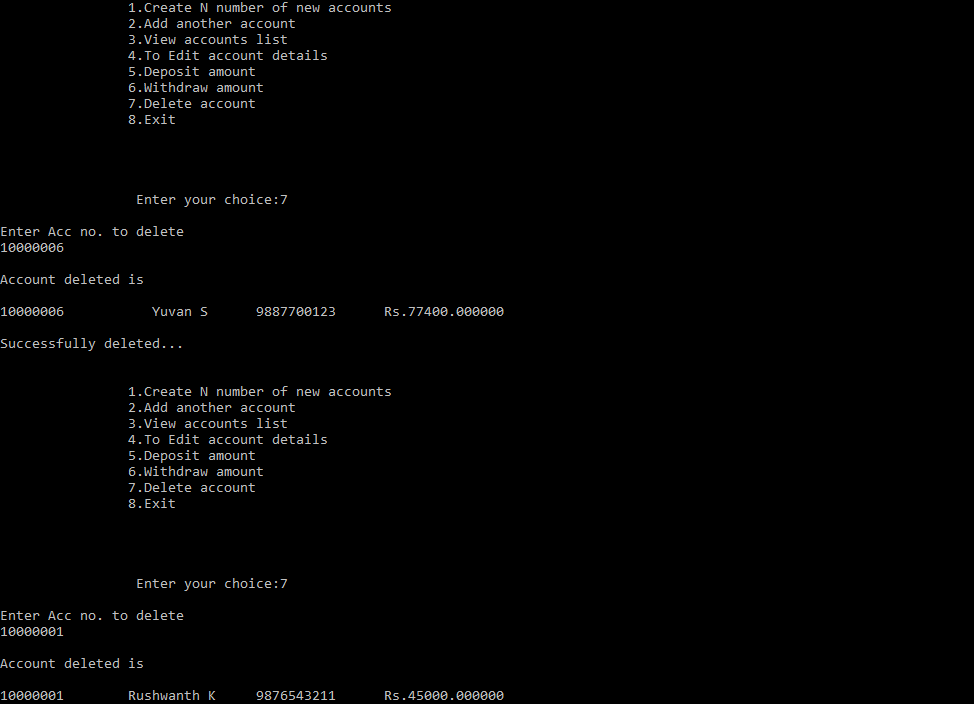
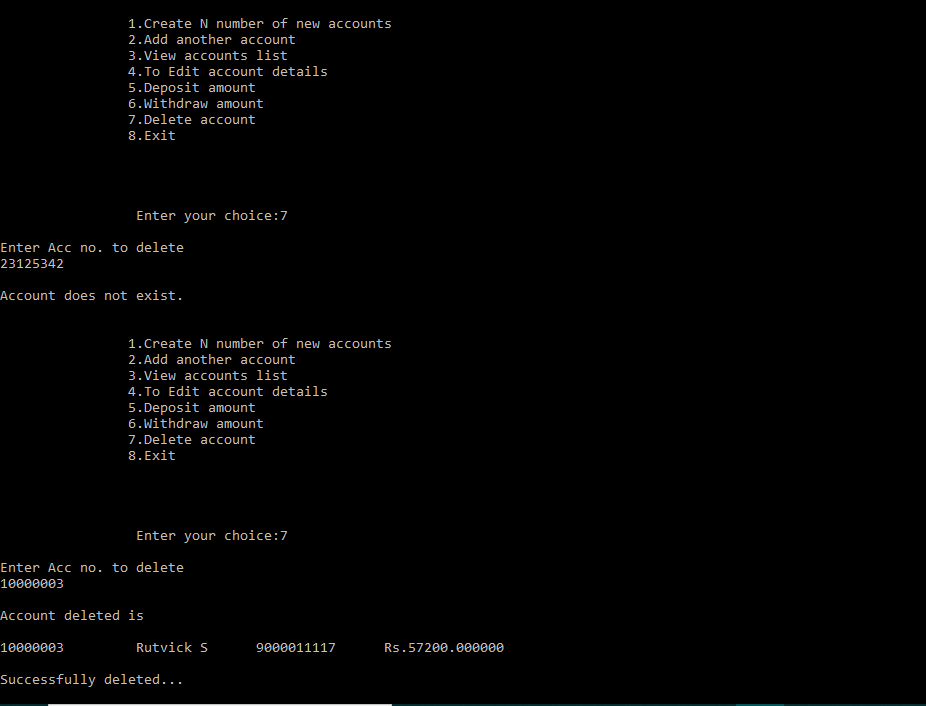
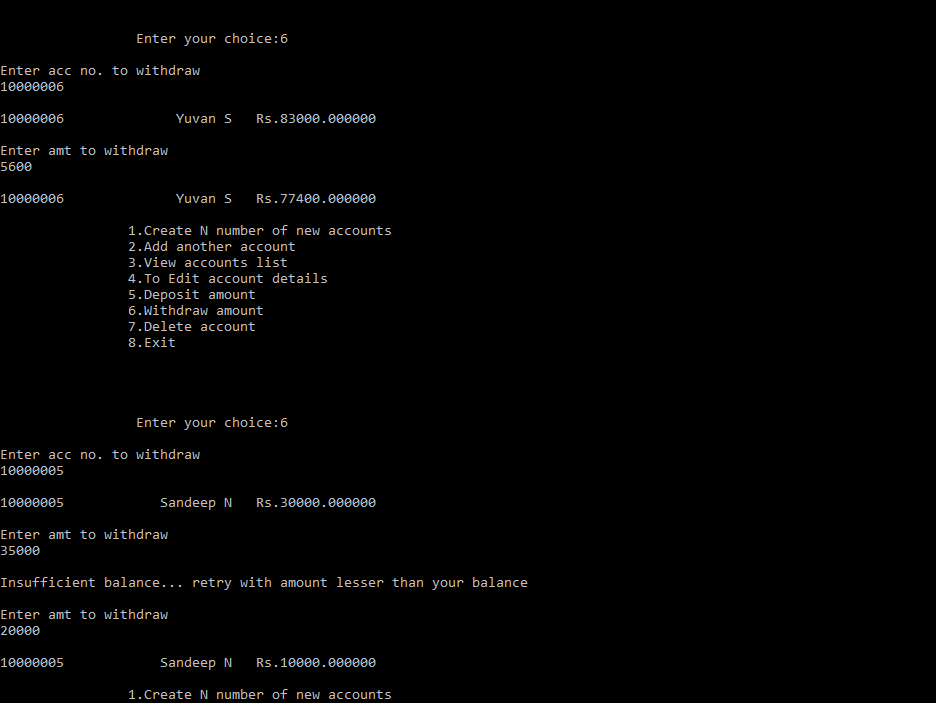
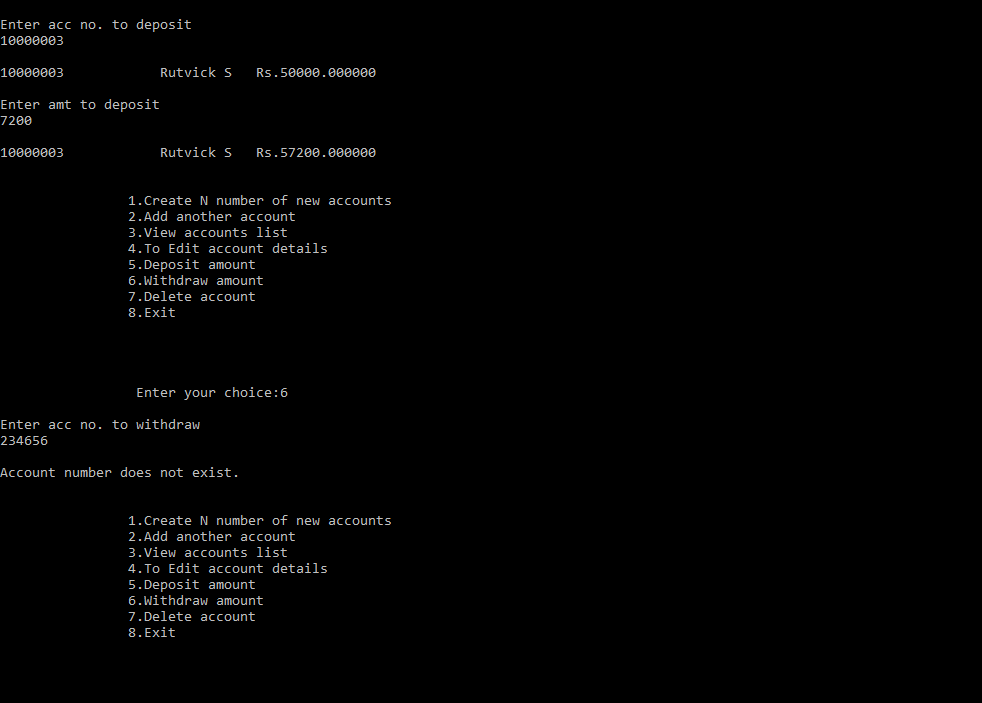
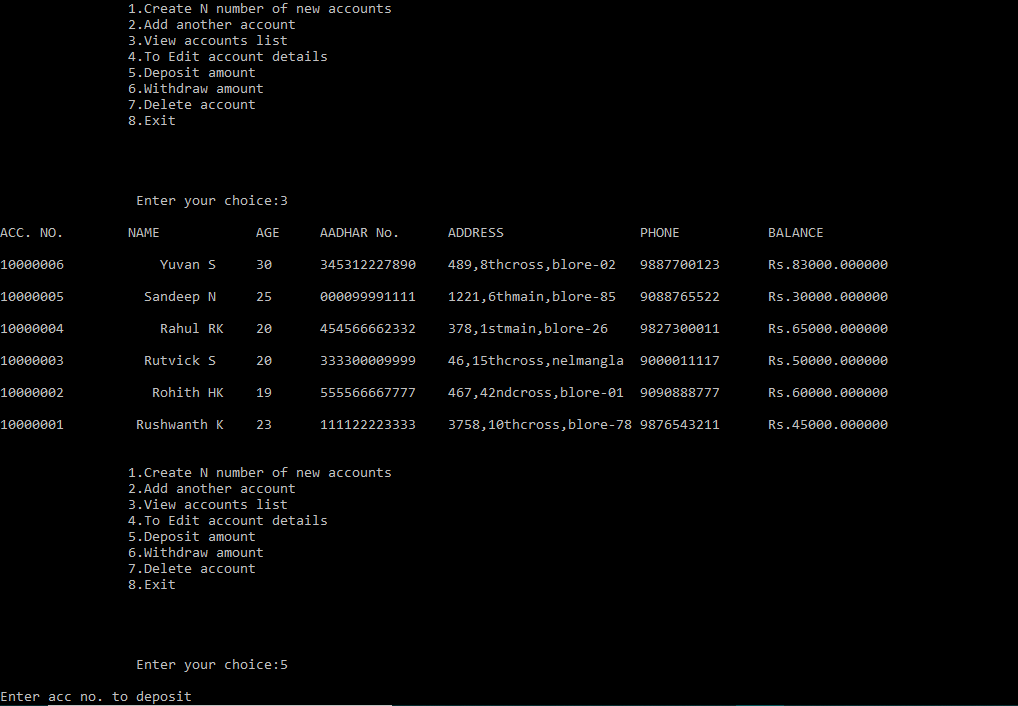
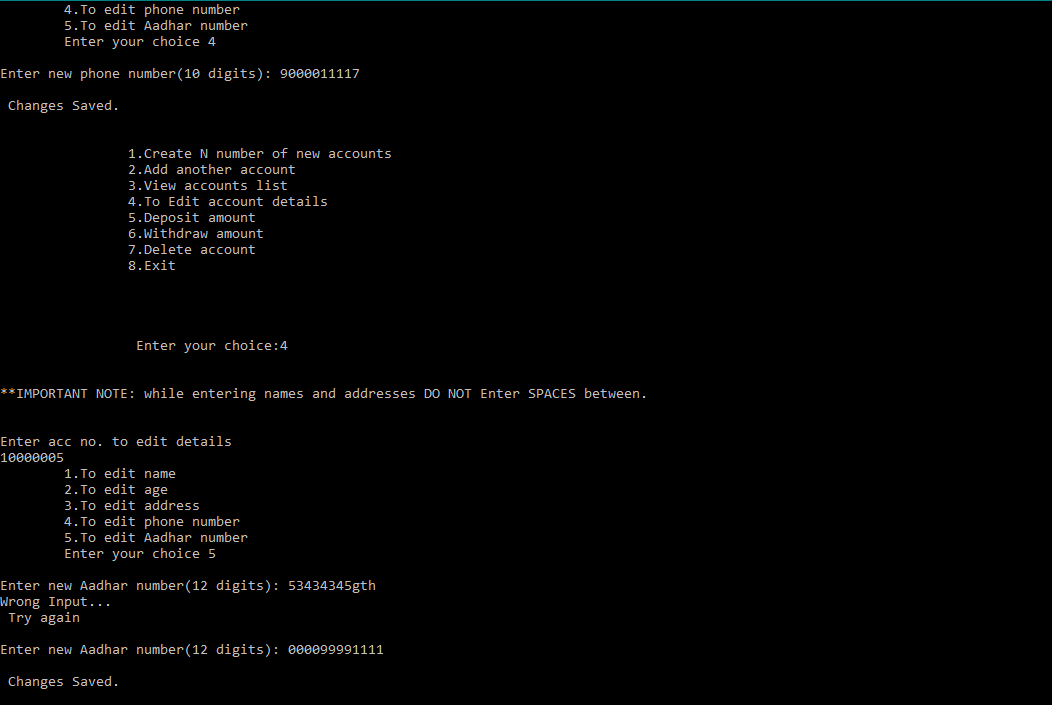
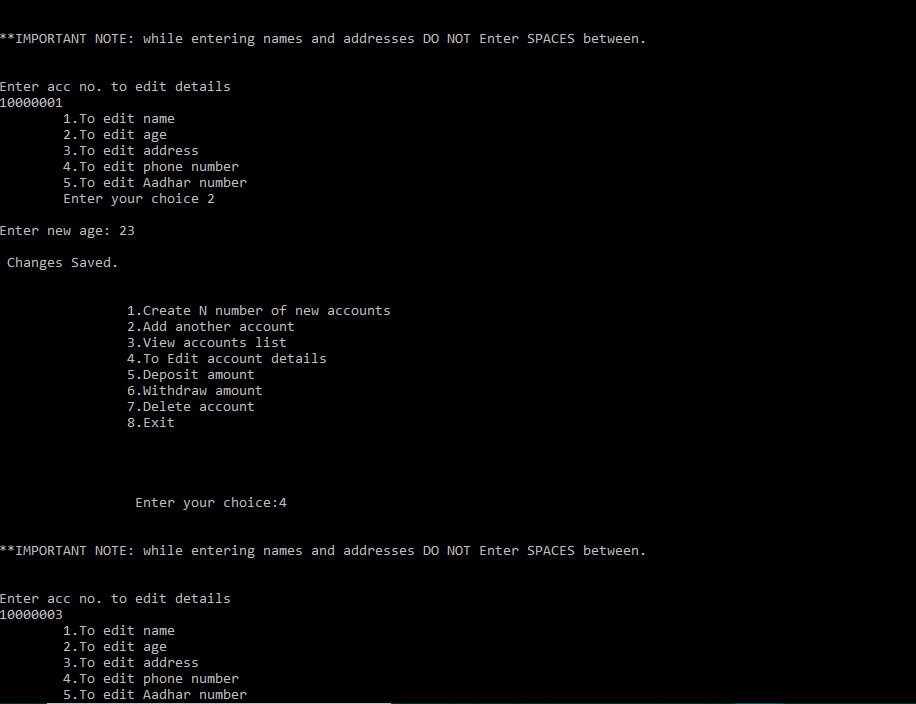
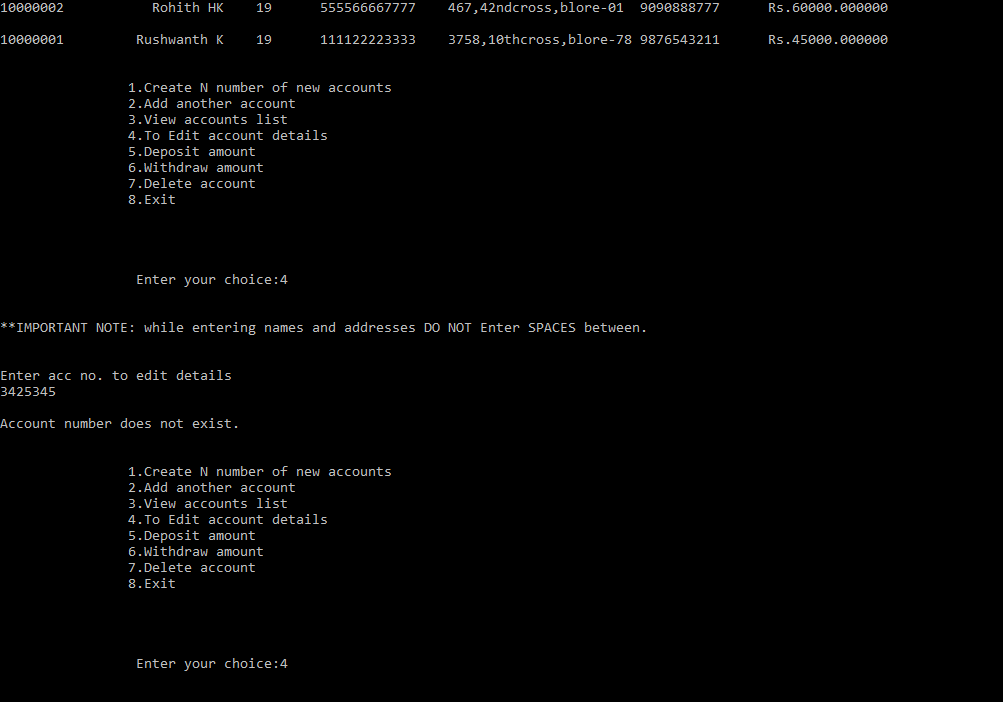
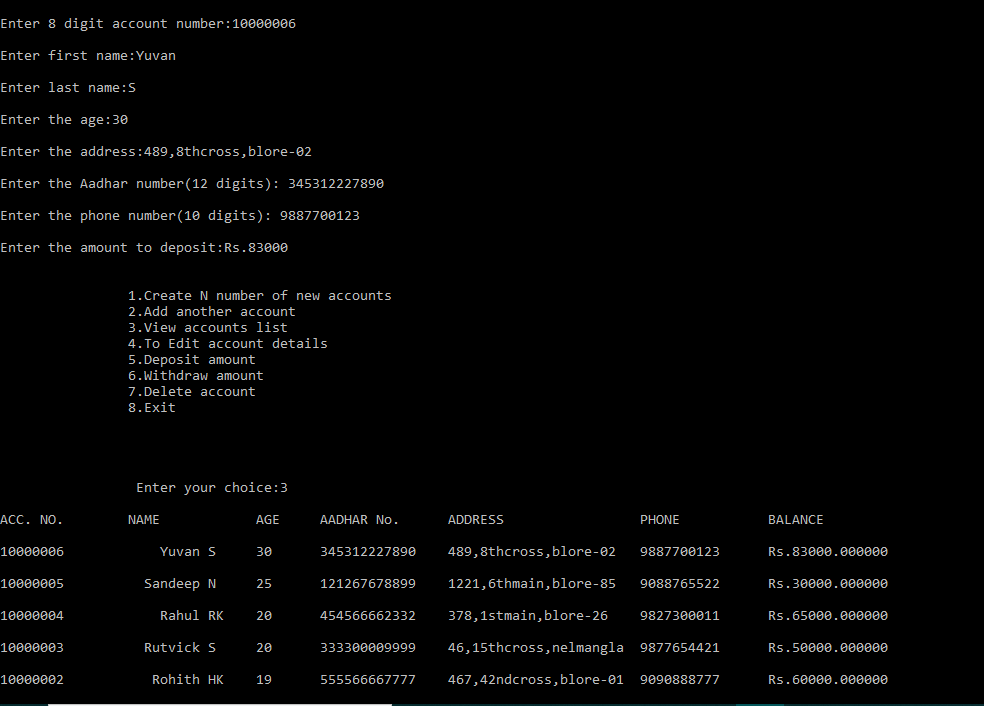
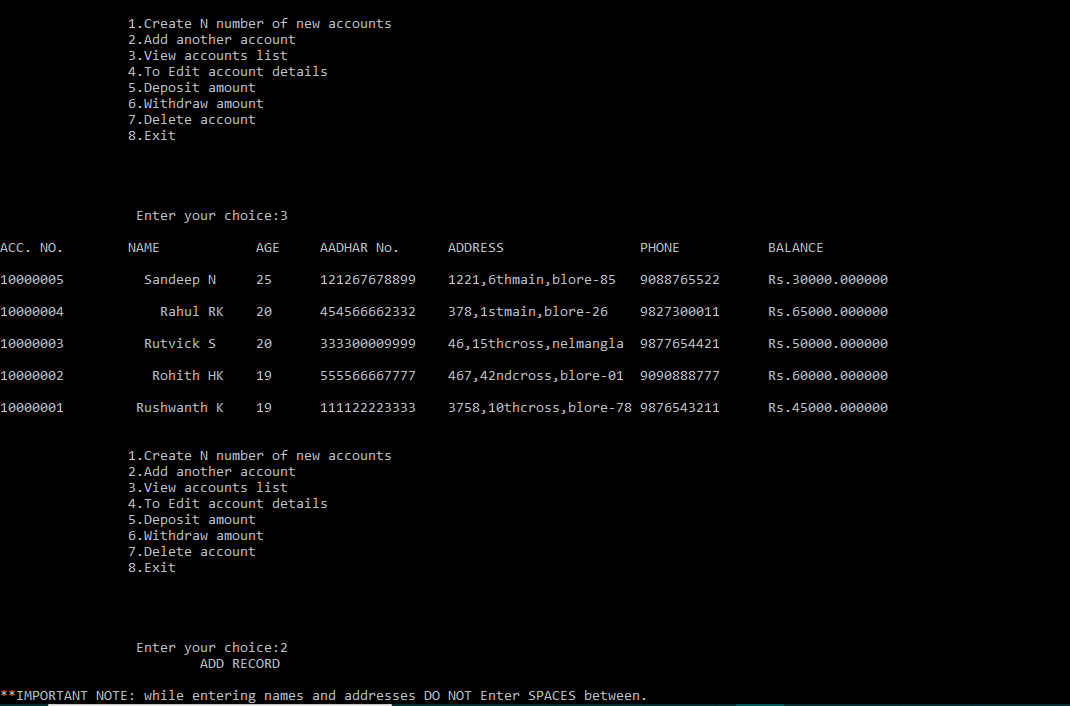
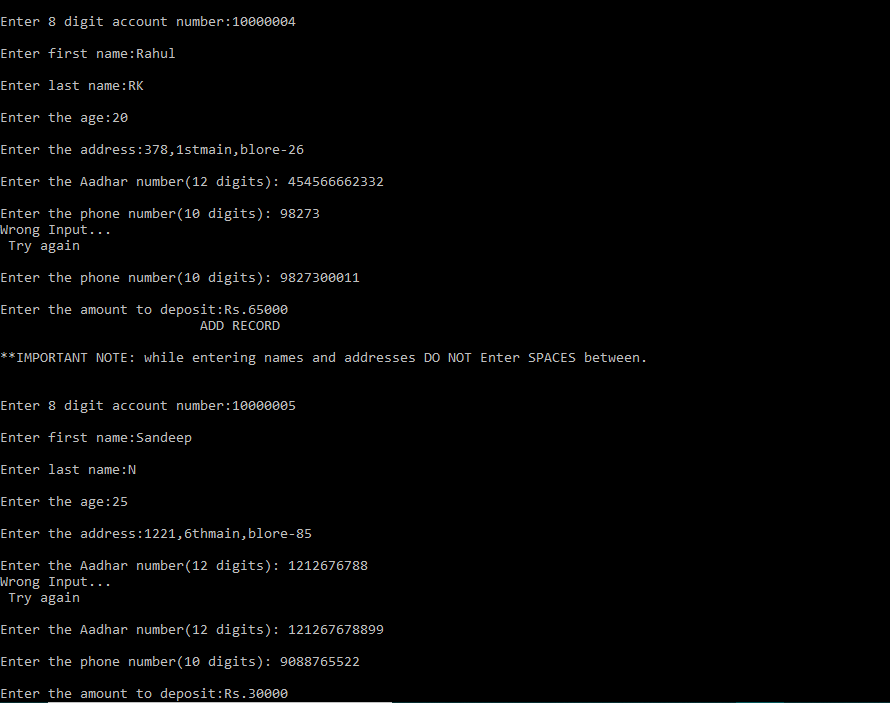
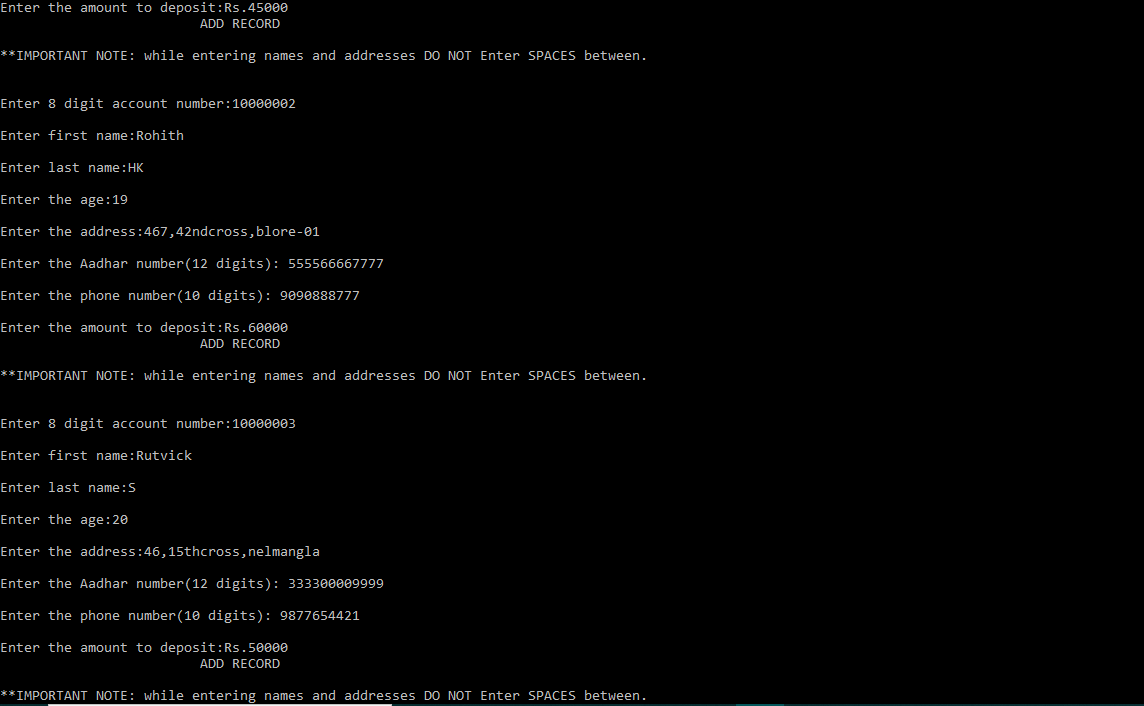
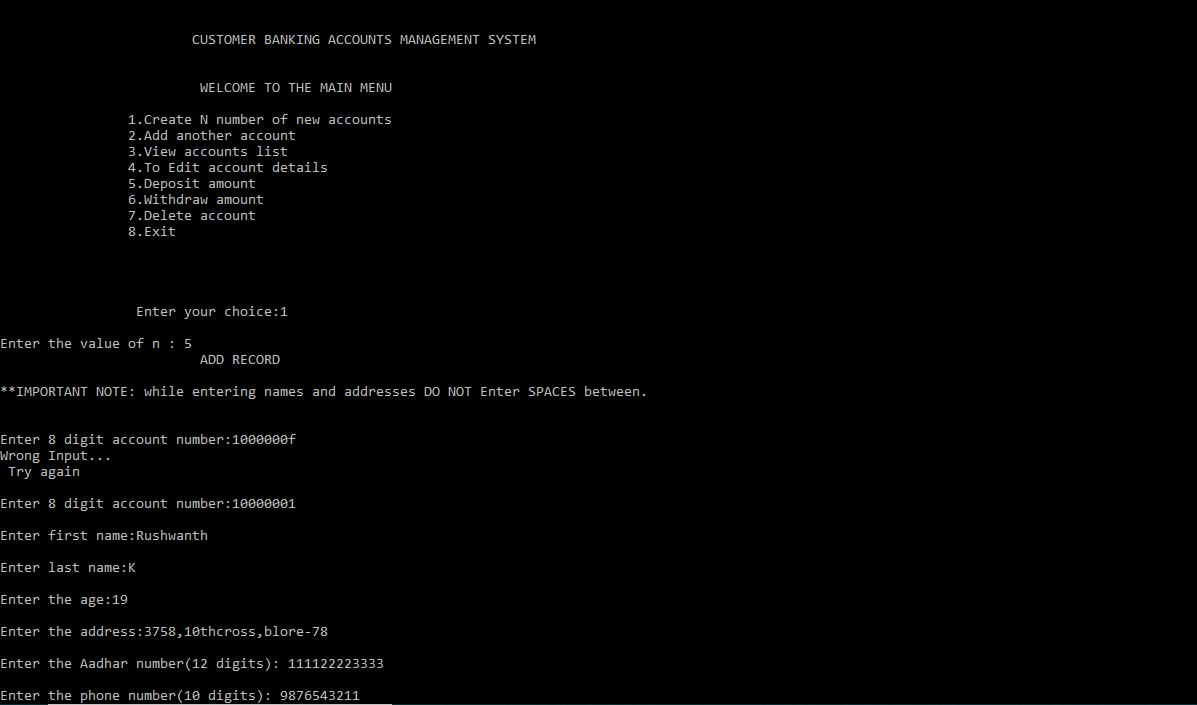
case 8: exit(1)

default: print ‘Incorrect choice’

end while

Step6: End.

**Results**

****

**Conclusion and Future Enhancements**

This banking system project will serve as a useful approach to data base dialog box to deposit and withdraw the money for the person. It will reduce the time taken by the user/admin to handle accounts. This program has a user friendly approach.

Currently handling savings account only in future we can handle other types of account ex:-Fixed deposit, retirement accounts, checking accounts, etc. And even online banking will be updated soon. Passwords and log-in Id’s will be added for better security and privacy.

**References**

* **Books-** ‘Programming in C & Data Structures’ by **A.A Godse**
* **Weblinks**- https://www.codewithc.com/mini-project-in-c-bank-management-system/