Bank Management System Implemented using Linked List

Outlines:

- All functionalities of linked list are available in this Assignment. That includes:
 - 1) Insert at beg, end, after, specific.
 - 2) Delete Node (Accounts closing and deleting fr list)
 - 3) Display List 4) Find a Node
 - 5) Number of Nodes 6) Sort List (Alphabetically) Ascending and descending. 7) Reversing List.
- 8) Get Node Data and manipulate it (All possible operation of Data(Account) of Node implemented). Apart from this

Account

- 1) Read and Write of list in file.
- 2) Both, Automatically and manual updating of file. 3) A Single Variable for all iterators & single array for Menu Inputs.

The System is can be used by bank manager or the account holder. For management area, the files can be imported and exported. List can be exported in two ways, one is difficult to understand by human(for backup with confidential info) and the other is easy to read & understand(such as Bank statement). Furthermore the list can be sorted alphabetically Ascending and descending. And list can also be reversed alphabetically.

The input to entire system is well validated.

-Name : char[100] -Account_Num: string -Balance : long double -isActive : bool

The user can perform all bank operations such as deposit, withdraw, and Transfer.

The system has advanced PIN feature, the PIN is entered with help of _getch(), its disadvantage is that it accepts all character even backspace. So in order to overcome this, the pin system has been designed to distinguishes between numbers, alphabets (blocks them), and backspace and so The backspace works perfectly fine, and the user can retype the pin.

UML Diagram:

```
-Minimum_Balance : int
                                       -Account_Type : char
                                       -PIN : char[5]
                                       +Account()
                                      +Account(char *, string, double : long, string, int, string, char *)
                                      +Account_Creation() : void
                                       +set_PIN(): void
                                       +set_Name(char *) : void
                                       +set_accountNo(AccountNum: string): void
                                      +set_balance(a : long double) : void
                                      +update_balance(a : long double) : void
                                       +setisActive(truefalse : bool) : void
                                      +set_MinBalance(Minimum_Balance : int) : void
                                       +set_accountType(char) : void
                                       +get(): void
                                       +get_AccountNo(): string
                                       +get_Account_Holder_Name() : char *
                                       +get_balance() : long double
                                       +get_minBalance(): int
                                       +is_current_or_saving_accountType(): bool
                                      +Deposit() : void
                                       +getPIN() : char *
                                       +Withdraw (): void
                                       +is_Active(): bool
                                      +Suffient_Balance(Amount : long double) : bool
                                       +~Account()
                                       +saving dedution(balance : long double) : void
                                       +==(Student_2 : Account) : bool
                                            Account_Data
                                                                                                                            list
                                                                                           head : Node*
                                                                                           -tail : Node*
                                                                                           +list()
                                                                                           +get_head() : Node *
                                         next
                                                                                           +get_tail() : Node *
                                                                                           +insert_beg(New Account : Account) : void
                                             +Account_Data : Account
                                                                                           +insert_specfic_Position(place : int, New Account : Account) : void
                                                                            ≪t<del>ai</del>l - - -
                                             +next : Node*
                                                                                           +insert_after(oldAccount : Account, New Account : Account) : void
                                             +Node()
                                                                                           +insert_end(New Account : Account) : void
                                                                               head
                                             +Node(New Account : Account)
                                                                                           +Display_list(): void
                                                                                           +isEmpty(): bool
                                                                                           +Number of Nodes():int
                                                                                           +delete_Account_Node_fr_list(Account_Pos_to_delete : Account) : void
                                                                                           +Display_Node(i : int) : void
                                                                                           +get_Account_Node(i : int) : Node *
Screen Shots of Program Working:
```

Welcome to Bank Account Management System 0. Manage Account DataBase 1. Create a Account

C:\Users\Wakeel Furqan Ahm 🔻

C:\Users\Wakeel Furqan Ahm X

If no accounts:

5. d 6. b

1. wger 2. sdfdsf

C:\Users\Wakeel Furqan Ahm X

List Before Reversing

Sort Alphabetically (asc)

3. q 4. f 5. d 6. b 7. b 8. a

1. g 2. e

newaccout

PIN Assigned:9712

0. Manage Account DataBase

Head:Wakeel

```
4. Manupulate List
                Export As Readable
                                        3. Export As Backup
  Import
 🖾 C:\Users\Wakeel Furqan Ahm 🗡
1. Import
                Export As Readable
                                        3. Export As Backup
                                                                 4. Manupulate List
Data has been successfully Imported
0. Manage Account DataBase
                                1. Create Account
                                                         2. Manage Accounts
3. Deposit Money
                                4. Withdraw Money
                                                         5. Transfer Funds
6. Exit Program
 C:\Users\Wakeel Furqan Ahm X
                2. Export As Readable
                                        3. Export As Backup
                                                                 4. Manupulate List
  Import
```

```
C:\Users\Wakeel Furqan Ahm 🔻

    Sort Alphabetically (asc)

                                Sort Alphabetically (desc)
                                                                 3.Reverse List
List Before Sorting
1. wqer
sdfdsf
3. q
4. f
```

```
7. b
8. a
Sorted List:
1. a
2. b
3. b
4. d
5. f
6. q
sdfdsf
8. wqer
                                1. Create Account
0. Manage Account DataBase
                                                        2. Manage Accounts
3. Deposit Money
                                4. Withdraw Money
                                                        5. Transfer Funds
6. Exit Program
 C:\Users\Wakeel Furqan Ahm X
1. Import
                Export As Readable 3. Export As Backup 4. Manupulate List

    Sort Alphabetically (asc)
    Sort Alphabetically (desc)

                                                                Reverse List
List Before Sorting
1. a
2. b
3. d
4. sdfdsf
5. wger
6. q
7. b
8. f
Sorted List:
```

```
4. Testaccount
qwerty
Reversed List:

    qwerty

2. Testaccount
newaccout
4. e
5. g
0. Manage Account DataBase
                                1. Create Account
                                                         2. Manage Accounts
3. Deposit Money
                                4. Withdraw Money
                                                         5. Transfer Funds
6. Exit Program
 C:\Users\Wakeel Furqan Ahm X
# of Accounts:0
Account Creation
Note Minimum Balance limit: Current Account: Rs.5000/- Saving Account: Rs.1000/-
Select Account Type
'C'. Current Account
'S'. Saving Account
AccountNo Assigned:ED06V74015
Enter Account Holder Name: Wakeel
Enter the Amount You Want To Add in account:10000
```

Tail:Wakeel

Sort Alphabetically (desc)

3.Reverse List

2. Manage Accounts

2. Manage Accounts

6. Exit Program

```
3. Deposit Money
                                  4. Withdraw Money
                                                             6. Exit Program
Deposit:
  ©:\ C:\Users\Wakeel Furqan Ahm : X
 Select Account To Deposit Money Into

    Wakeel

 0. CancelEnter Amount to Deposit:500
 Transaction Successful !!
 0. Manage Account DataBase

    Create Account

                                                             2. Manage Accounts
 Deposit Money
                                   4. Withdraw Money
                                                             6. Exit Program
Withdraw:
Transfer:
No Transfer available if 1 account exists:
```

1. Create Account

Select Account To Deposit Money Into Wakeel

0. Manage Account DataBase Deposit Money

Transaction Successful !!

6. Exit Program

© C:\Users\Wakeel Furqan Ahm × + ∨

0. CancelEnter Amount to Deposit:500

```
More than 1 account then only transfer function is available:
  🖎 C:\Users\Wakeel Furqan Ahm 🗡
# of Accounts:1
Account Creation
Note Minimum Balance limit: Current Account: Rs.5000/- Saving Account: Rs.1000/-
Select Account Type
 'C'. Current Account
 'S'. Saving Account
AccountNo Assigned:B559751864
Enter Account Holder Name:newaccount
Enter the Amount You Want To Add in account:13231
PIN Assigned:8907
Manage Account DataBase
                                  1. Create Account
                                                           2. Manage Accounts
```

Create Account

4. Withdraw Money

```
Transfer Funds
3. Deposit Money
                               4. Withdraw Money
6. Exit Program
    C:\Users\Wakeel Furqan Ahm 🔻
   Select Account To Transfer Funds FROM:

    wakeel

   2. newaccount
   Enter the 4-digit PIN to perform Transaction:7808
   Select Account To Transfer Amount TO:Enter Amount To Transfer:2320
   Transaction is Sucessful
   0. Manage Account DataBase 1. Create Account
                                                           2. Manage Accounts
   3. Deposit Money
                                   4. Withdraw Money
                                                           5. Transfer Funds
```

NOTE THE PIN DISPLAYED is for showing purpose only.