SdPd/java Lab Exam 3

Objective: City Tribes Bank

The City Tribes bank maintains account data using sequential text files and arrays.

1. **Download** the lab exam 3 **zip** file and extract the folder, **Save** on the desktop (**not** your Network account, local C drive or USB)

   – Rename the **LastNameFirstNameLabEx3** folder and starter java file as per your own name
   – E.g. **AgnewGerryLabEx3** folder and **AgnewGerryLabEx3**.java file
   – To be **verified** by your lab supervisor

2. Add your **Program Id, Name** and **Program Description** as **comments** at the top of the java program

3. Remember to rename the starter **class name** as your java file name then **Save, Compile and Run** the program **before** you write any further code
   – Alert your supervisor if the program does not run

4. **Warning:**
   – Marks will be deducted for **bad programming practices** such as:
   – Lacking meaningful variable names, white-space, indentation, etc.
   – Ensure redundant code is deleted prior to program submission

5. **Account File layout:**
   Each record consists of the following details about each bank account:
   – accNo (integer) – unique 4 digit customer number e.g. 1234
   – accOverdrawn (double) – account overdraw limit e.g. -200.99
   – accBalance (double) – account balance e.g. 111.22
   – accType (char) – account type; S➔Savings, C➔Current, I➔Invest
   – accName (String) – account holder name e.g. Gerry Agnew
   – See the account file contents attached – Screenshot 1

6. **Tx File layout:**
   Each record consists of the following details about each transaction:
   – txNo (integer) – unique 4 digit account number e.g. 1234
   – txAmount (double) – transaction amount e.g. 111.22
   – txType (char) – L ➔ Lodgement and W ➔ Withdrawal
   – See the tx file contents attached – Screenshot 1

7. **Sample Data:**
   – Number of Bank Account & Transaction records are unknown (max 50)
   – Both files are in ascending account number order and terminated with dummy 9999/EOF/sentinal account number records
   – Verify the contents of both input text files using NotePad

8. **Constants:**
   Declare any constants required with meaningful names/types

9. **Variables/Arrays:**
   Declare file objects, file variables, arrays & other variables as appropriate with meaningful names/types

10. **Initialise:**
    Initialise any necessary variables especially Counts, Totals and Booleans

11. **Header Output:**
    – Output the program header including **your name** as specified
    – See the attached Screenshot 2

12. **Account File Input to Arrays:**
    – Using an 9999/EOF controlled **while** loop read each bank account from the text file using the Initial/while/Subsequent read approach
    – Store the data fields in multiple appropriately named/typed **1-dim** arrays
    – Line output formatted bank account details using the **arrays** to verify that they have been correctly populated
    – Remember to trim leading/trailing spaces from the Account Name string
    – See the attached Screenshot 2

13. **Tx File File Input:**
    – Using an 9999/EOF controlled **while** loop read each tx record from the file using the Initial/while/Subsequent read approach
    – Line output formatted tx details to verify that the transactions are being correctly read (initially only Tx number, Amount and Type)
    – See the attached Screenshot 2

14. **Tx File Processing:**
    – Modify the above Initial/while/Subsequent read to find the tx account number in the associated array
    – Then add/subtract the tx amount to/from the account balance based on the txType e.g. add lodgements and subract withdrawals
    – Line output formatted account info to verify that the transactions are being correctly processed including the after Tx Account balance
    – Mismatched Tx account numbers (in tx file & not in the Account number array) should be reported using the Mismatched Tx Report as indicated
    – See the attached Screenshots 2 & 3

15. **Account Name Processing:**
    – Using an outer **while** loop prompt for the search Account name using the Initial/while/Subsequent read approach
    – Until “Stop” is entered (case insensitively) to end the search
    – See the attached Screenshot 4

16. **Account Name Search:**
    – Using an inner **while** loop search the Account Names array case insensitively for the Account Name entered
    – Using at least one boolean variable

– Where unsuccessful output an appropriate error message
– See the attached Screenshot 4

17. **Successful Account Name Search Processing:**
    – If successful input/prompt for the Account amendment character option N/ame, L/odge, W/ithdraw, A/ccount Info, O/verdrawn, I/gnore using a switch statement
    – Show the current value e.g. Current name: Gerry Agnew
    – Amend the corresponding array element with the new value entered
    – For option A: Output/display all Account info using the same format as before with a for loop
    – For option O: Output/display all Overdrawn Accounts info (negative balance) using the same format as before with a for loop
    – See the attached Screenshots 4 & 5

18. **Exception Validation:**
    – Validate case insensitive character option – must be N/L/W/A/O/I using **switch** validation
    – Validate the customer Name – cannot be blank (hint trim) using **do … while** validation
    – Validate the Lodge/Withdraw amounts – min 0.01 and max 1000.00 using **do … while** validation
    – Withdrawals are not permitted which exceed the overdrawn limit

19. **Output Updated Arrays to File:**
    – When finished output the modified array contents to a new text file called "**NewBankAccount.dat**" verified using NotePad
    – Remember to add the dummy 9999/EOF/sentinal record at the file end
    – See the attached Screenshot 6

20. **Close Files:**
    Close the file objects

21. **Save – The End:**
    – When finished Save and Exit TextPad
    – Zip (R/click: Send Compressed) the **LastNameFirstNameLabEx3** folder
    – Upload the **LastNameFirstNameLabEx3** zip file to Moodle link provided
    – To be **verified** by your supervisor **before** you **submit** the zip file
    – Remember to submit your "Named" Algorithm sheet and any rough work
    – Sign the **attendance sheet** before you exit the lab

Screenshot 1

BankAccount.dat - Notepad

File Edit Format View Help

```
1001   -100.00    100.01   S Gerry Agnew
2002   -300.00     50.02   I Mary Murphy
3003   -400.00   -300.03   C Gerry Murphy
4004  -1000.00   4000.04   S Jane Doe
5005      0.00     55.05   S Felix Agnew
6006   -200.00    100.06   I John Doe
7007      0.00    170.07   C Paddy Reilly
9999      0.00      0.00     End of File
```

BankTx.dat - Notepad

File Edit Format View Help

```
1001      10.00  L
1001      10.00  W
1001     100.00  L
3333     100.00  L
2002     200.00  L
2002     200.00  L
6006     300.00  L
6006     100.00  W
8888     100.00  W
9999       0.00  End of File
```

Screenshot 2

C:\WINDOWS\system32\cmd.exe

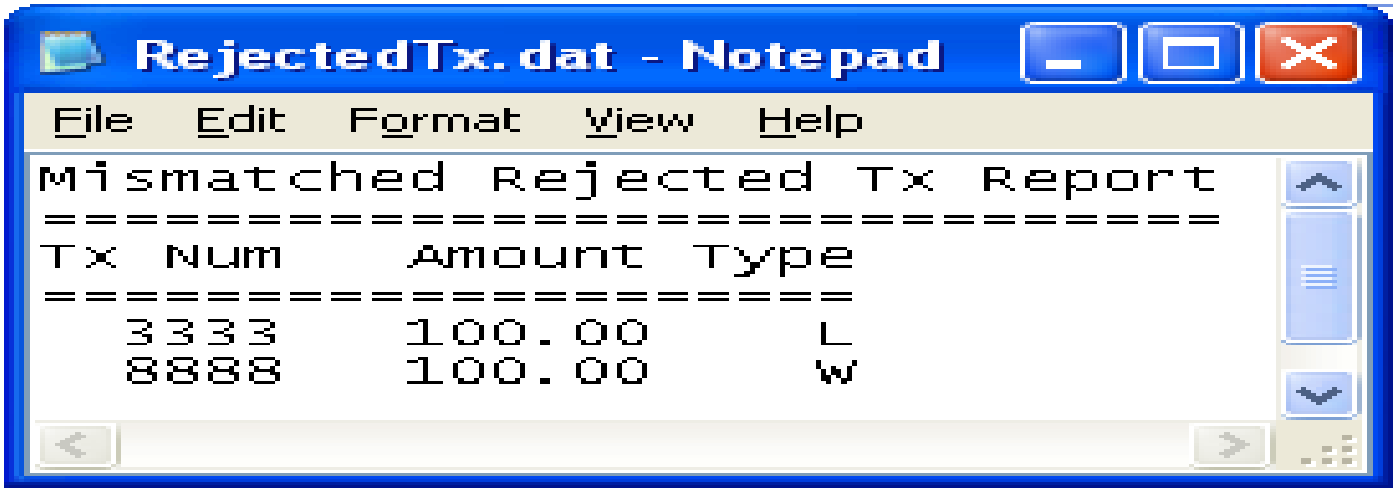
```
Gerry Agnew - Bank Account & Tx Processing

Acc No    Limit   Balance Type Account Name
=========================================
  1001  -100.00    100.01    S Gerry Agnew
  2002  -300.00     50.02    I Mary Murphy
  3003  -400.00   -300.03    C Gerry Murphy
  4004 -1000.00   4000.04    S Jane Doe
  5005     0.00     55.05    S Felix Agnew
  6006  -200.00    100.06    I John Doe
  7007     0.00    170.07    C Paddy Reilly
```

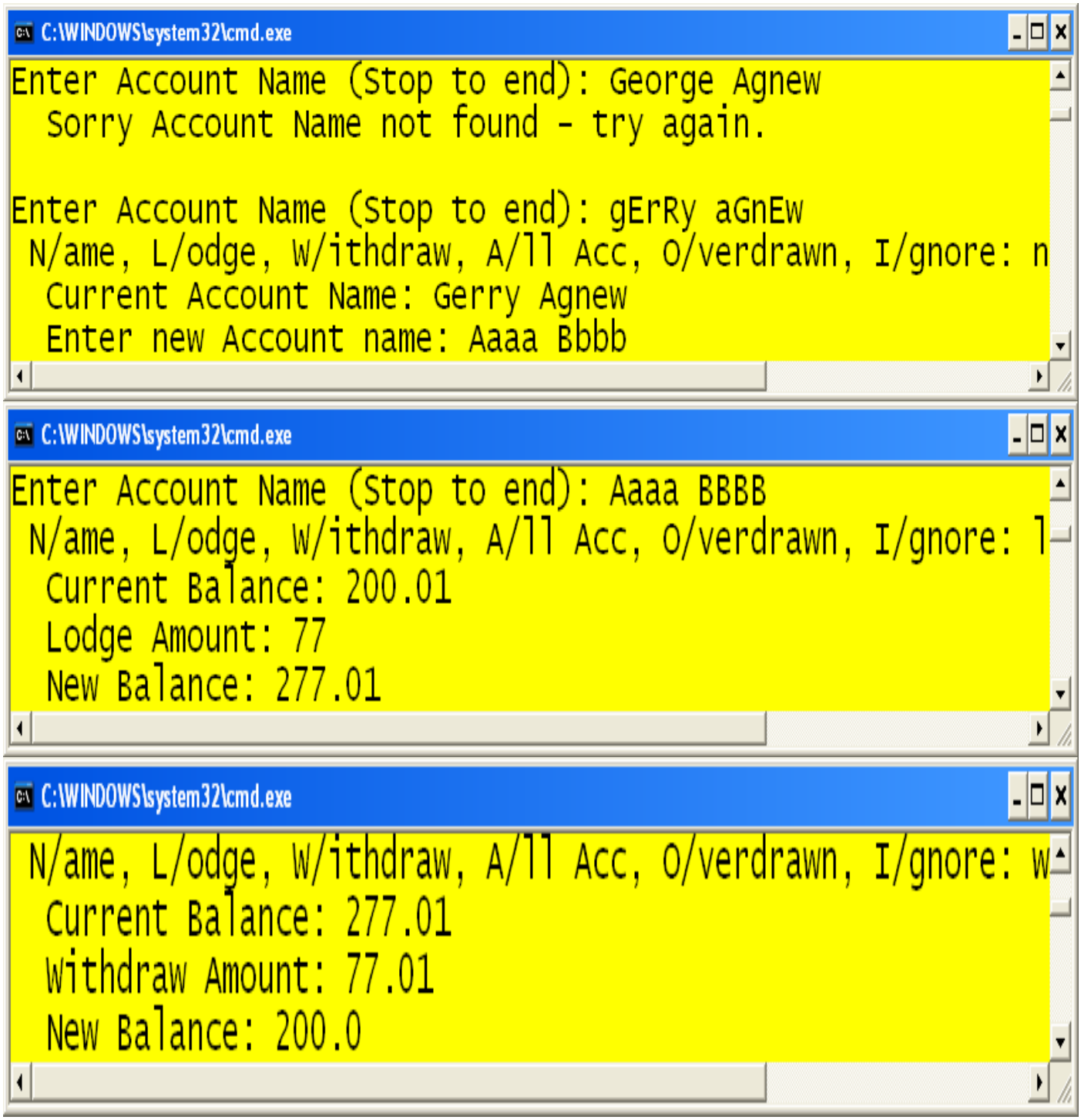
C:\WINDOWS\system32\cmd.exe

```
Tx Num    Amount Type Account Name     After Bal
=================================================
  1001     10.00    L Gerry Agnew          110.01
  1001     10.00    W Gerry Agnew          100.01
  1001    100.00    L Gerry Agnew          200.01
  2002    200.00    L Mary Murphy          250.02
  2002    200.00    L Mary Murphy          450.02
  6006    300.00    L John Doe             400.06
  6006    100.00    W John Doe             300.06
```

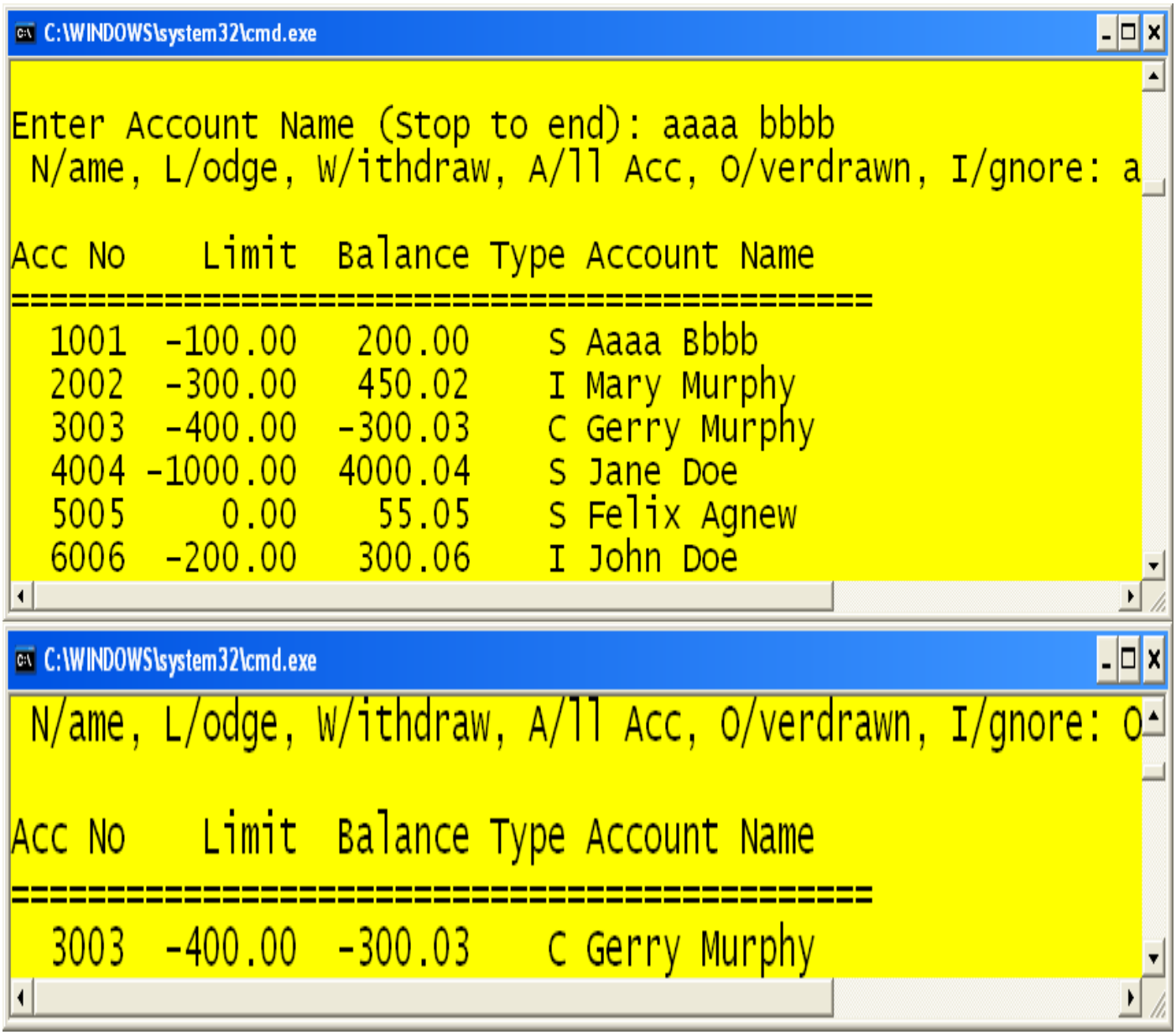
Screenshot 3



Screenshot 4



Screenshot 5



Screenshot 6