Lab 4 in TND002

Mark Eric Dieckmann

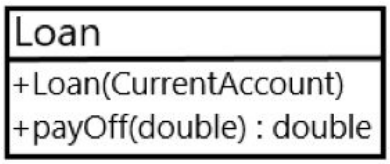
February 15, 2023

1 Summary

You expand the code framework you started to implement in lab 3. The new features are the possibility to send money between current accounts, to take loans, and to add interest to the balance on the accounts. For this purpose, you create the new class **Loan** and you expand your existing classes. I will mark the variables and methods, which need to be introduced or updated, in blue in the class diagrams.

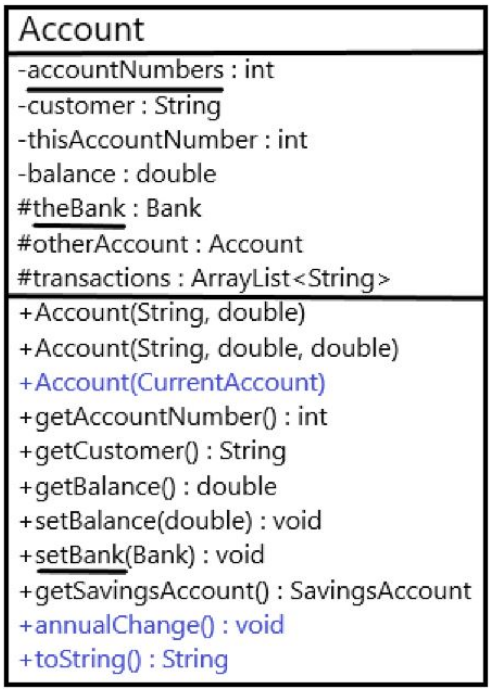
2 Task: Implement the new class Loan

This is a subclass of **Account**. Its constructor takes a current account as its argument. It passes this argument to a constructor of **Account**, which is not yet implemented.



payOff(arg) reduces the loan by *arg*. The balance on a loan is negative so you add the positive *arg* to reduce it. The method adds a string "Paid off: " followed by the value of *arg* to its *transactions* array. It returns the updated value of *balance*. A positive return value will cause this loan to be deleted by the bank and the returned money will be used to pay off more loans of that customer (if there are any). Thus, you do not need to worry about a loan with a positive *balance*.

3 Task: Expand the class Account



You add one more constructor *Account(arg)* that allows you to initialize instances of **Loan**. It initializes *otherAccount* with *arg* and *balance* with the balance of *arg*, which will be negative-hence the loan. It initializes *thisAccountNumber* with that of *arg* and *customer* with that of *arg*.

You also add the method *annualChange()*, which changes the value of *balance* based on the account type.

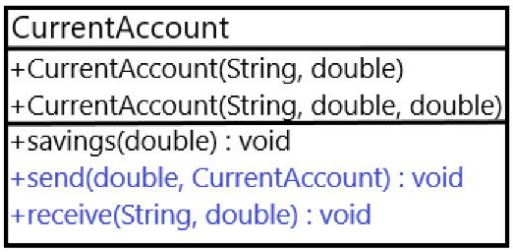
If the account is a savings account, it should add 1% interest to *balance*.

If the account is a loan, it should add 5% interest to *balance* (*balance* is negative).

If the account is a current account, it should subtract 10 from *balance*. The value of *balance* in the current account should not become negative. If it becomes negative, the current account should call the *getLoan(arg)* method of **Bank** with itself and set the value of *balance* to zero.

Until now, *toString()* distinguishes only between current accounts and savings accounts. Expand it such that it can also deal with loans. It should start the line with the string "Loan:" followed by the value of *balance* followed by the transactions involving this loan.

4 Task: Expand the class CurrentAccount



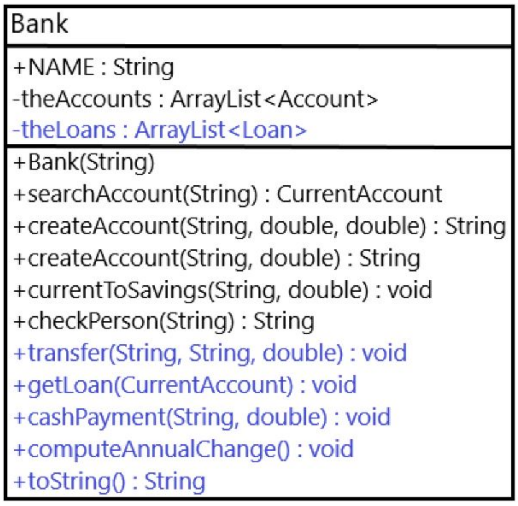
receive(arg1, arg2) allows you to send money to the calling account. Add *arg2* to *balance* of the calling account. If *arg1* is "Cash payment", then you add "Cash payment: " followed by the value of *arg2* to *transactions* of the calling account. Otherwise, the money comes from another current account. In this case, *arg1* is the name of the holder of the account where the money comes from. You add "Received from account of " followed by the value of *arg1* followed by ": " followed by the value of *arg2*.

send(arg1, arg2) allows you to send the amount *arg1* from this account to the current account *arg2*. The full amount has to be sent and you should add to *transactions* the

string "Sent to account of " followed by the name of the account holder of *arg2* followed by ": " followed by *arg1*. Subtract *arg1* from *balance* of the calling current account and call the receive(..) method of *arg2* with the value of *arg1* and the name of the customer that holds the account that is sending the money.

If *balance* on this account is negative after you sent the money to *arg2*, you collect money from the savings account (if this account has one) by calling the *savings(arg)* method. You either get enough money from the savings account to bring *balance* to zero or you get all of it if there is not enough money in the savings account to get *balance* to zero. If this current account still has a value *balance* below zero after the transfer from the savings account, you must take a loan to cover the difference. You call the *getLoan(arg)* method of the bank and add the string "Covered by a loan: " followed by the amount of money you got from the loan to get *balance* to zero. After taking the loan, *balance* should be set to zero on this current account.

5 Task: Expand the class Bank



theLoans is an array that stores all loans, which were given by the bank. You initialize it when you declare it.

transfer(arg1, arg2, arg3) transfers *arg3* from the account of the customer *arg1* to that of the customer *arg2* if both customers have a current account. It should do nothing if at least one of the customers does not exist. The method transfers the money by calling the *send(..)* method of the current account of *arg1* with the current account of the customer *arg2*. *receive(..)* is already called in the *send(..)* method and you do not need to call it here again.

getLoan(arg) creates an instance of **Loan** with the information stored in *arg* and adds it to *theLoans*. Everything else is done in the constructor of **Account**.

cashPayment(arg1, arg2) allows you to pay off loans of the customer *arg1* using the amount *arg2*. You go through the list of loans that were taken by the person with the name *arg1*. You pick a loan and call its *payOff(..)* method. If the return value of this method is positive you paid the loan off. You delete the paid-off loan and you use the return value to pay off another loan of the person *arg1*. Once the return value of *payOff(..)* becomes negative, you can no longer pay off loans. If you went through all loans and you still have money left over, you pay that into the current account of *arg1* via its *receive(..)* method and state "Cash payment" as the source.

computeAnnualChange() goes through all elements of *theAccounts* and *theLoans* and calls their *annualChange()* methods.

toString() has to be expanded. It should state the total number of loans in the same form as you stated the total number of accounts in lab 3. It should also sum up all the money in the loans and add that to the string (see console output).

6 Console output

The order of the sequence of snapshots is from left to right and top to bottom. The complete sequence can be found in "Output.txt".

```
Setting up the accounts
Current and savings accounts created
Current and savings accounts created
Current and savings accounts created
Current account created
Bank: Great Northern Bank
Accounts: 7
Loans: 0
Money in current / savings accounts and debt: 8000.0 / 6000.0 / 0.0

Transfer below maximum of current account

Peter

Current Account: 1000.0
Sent to account of Sofia: 1000.0

Savings Account: 2000.0

Sofia

Current Account: 3000.0
Received from account of Peter: 1000.0

Savings Account: 2000.0


Transfer of remainder of current account

Peter

Current Account: 0.0
Sent to account of Sofia: 1000.0
Sent to account of Sofia: 1000.0

Savings Account: 2000.0
```

```
Transfer of more money than on current account

Peter

Current Account: 0.0
Sent to account of Sofia: 1000.0
Sent to account of Sofia: 1000.0
Received from account of Sofia: 100.0
Sent to account of Sofia: 1000.0
From savings account: 900.0

Savings Account: 1100.0
To current account: 900.0


Transfer of more money than on current + savings accounts

Peter

Current Account: 0.0
Sent to account of Sofia: 1000.0
Sent to account of Sofia: 1000.0
Received from account of Sofia: 100.0
Sent to account of Sofia: 1000.0
From savings account: 900.0
Sent to account of Sofia: 1500.0
From savings account: 1100.0
Covered by a loan: 400.0

Savings Account: 0.0
To current account: 900.0
To current account: 1100.0

Loan: -400.0

Sofia

Current Account: 6400.0
Received from account of Peter: 1000.0
Received from account of Peter: 1000.0
Sent to account of Peter: 100.0
Received from account of Peter: 1000.0
Received from account of Peter: 1500.0

Savings Account: 2000.0
```

```
Bank: Great Northern Bank
Accounts: 7
Loans: 1
Money in current / savings accounts and debt: 10400.0 / 4000.0 / -400.0

Taking the second loan

Peter

Current Account: 0.0
Sent to account of Sofia: 1000.0
Sent to account of Sofia: 1000.0
Received from account of Sofia: 100.0
Sent to account of Sofia: 1000.0
From savings account: 900.0
Sent to account of Sofia: 1500.0
From savings account: 1100.0
Covered by a loan: 400.0
Sent to account of Sofia: 1500.0
Covered by a loan: 1500.0

Savings Account: 0.0
To current account: 900.0
To current account: 1100.0

Loan: -400.0

Loan: -1500.0

Sofia

Current Account: 7900.0
Received from account of Peter: 1000.0
Received from account of Peter: 1000.0
Sent to account of Peter: 100.0
Received from account of Peter: 1000.0
Received from account of Peter: 1500.0
Received from account of Peter: 1500.0

Savings Account: 2000.0

Bank: Great Northern Bank
Accounts: 7
Loans: 2
Money in current / savings accounts and debt: 11900.0 / 4000.0 / -1900.0
```

```
Testing debt for account without savings account

Alex

Current Account: 0.0
Sent to account of Sofia: 5000.0
Covered by a loan: 3000.0

Loan: -3000.0

Sofia

Current Account: 12900.0
Received from account of Peter: 1000.0
Received from account of Peter: 1000.0
Sent to account of Peter: 100.0
Received from account of Peter: 1000.0
Received from account of Peter: 1500.0
Received from account of Peter: 1500.0
Received from account of Alex: 5000.0

Savings Account: 2000.0
```

```
Pay off Peter's loans

Peter

Current Account: 0.0
Sent to account of Sofia: 1000.0
Sent to account of Sofia: 1000.0
Received from account of Sofia: 100.0
Sent to account of Sofia: 1000.0
From savings account: 900.0
Sent to account of Sofia: 1500.0
From savings account: 1100.0
Covered by a loan: 400.0
Sent to account of Sofia: 1500.0
Covered by a loan: 1500.0

Savings Account: 0.0
To current account: 900.0
To current account: 1100.0

Loan: -400.0

Loan: -1500.0

Peter

Current Account: 0.0
Sent to account of Sofia: 1000.0
Sent to account of Sofia: 1000.0
Received from account of Sofia: 100.0
Sent to account of Sofia: 1000.0
From savings account: 900.0
Sent to account of Sofia: 1500.0
From savings account: 1100.0
Covered by a loan: 400.0
Sent to account of Sofia: 1500.0
Covered by a loan: 1500.0

Savings Account: 0.0
To current account: 900.0
To current account: 1100.0

Loan: -100.0
Paid off: 300.0

Loan: -1500.0
```

```
Peter

Current Account: 0.0
Sent to account of Sofia: 1000.0
Sent to account of Sofia: 1000.0
Received from account of Sofia: 100.0
Sent to account of Sofia: 1000.0
From savings account: 900.0
Sent to account of Sofia: 1500.0
From savings account: 1100.0
Covered by a loan: 400.0
Sent to account of Sofia: 1500.0
Covered by a loan: 1500.0

Savings Account: 0.0
To current account: 900.0
To current account: 1100.0

Loan: -1300.0
Paid off: 200.0

Peter

Current Account: 100.0
Sent to account of Sofia: 1000.0
Sent to account of Sofia: 1000.0
Received from account of Sofia: 100.0
Sent to account of Sofia: 1000.0
From savings account: 900.0
Sent to account of Sofia: 1500.0
From savings account: 1100.0
Covered by a loan: 400.0
Sent to account of Sofia: 1500.0
Covered by a loan: 1500.0
Cash payment: 100.0

Savings Account: 0.0
To current account: 900.0
To current account: 1100.0

Bank: Great Northern Bank
Accounts: 7
Loans: 1
Money in current / savings accounts and debt: 15000.0 / 4000.0 / -3000.0
```

```
Testing annualChange() on current and savings accounts

Olga

Current Account: 2000.0

Savings Account: 2000.0

Olga

Current Account: 1990.0

Savings Account: 2020.0

Olga

Current Account: 0.0
Sent to account of Sofia: 1990.0

Savings Account: 2020.0

Olga

Current Account: 0.0
Sent to account of Sofia: 1990.0

Savings Account: 2040.2

Loan: -10.5
```