Logic coverage

CurrentAccount Class

1. balance < 500 {T, F}

SavingsAccount Class

1. amount < maxWithLimit {T, F}

BankAccount Class

1. (balance-amount) >= min_balance && amount < balance

2 clauses:

1. (balance-amount) >= min_balance
2. amount < balance

CoC – Combinatorial Coverage

(balance-amount) >= min_balance	amount < balance	Predicate
T	T	T
T	F	F
F	T	F
F	F	F

Test cases

TT: balance = 1500, amount = 200, min_balance = 500

TF: balance = 1000, amount = 1000, min_balance = 0

FT: balance = 2000, amount = 500, min_balance = 2000

FF: balance = 2000, amount = 3000, min_balance =500

Bank Class

addAccount method

1. getAccounts()[i] == null {T, F}

findAccount method

2. getAccounts()[i] == null
3. getAccounts()[i].acc_num.equals(accountNum) {TF, FT, FF}

deposit method

4. amt < 0
5. temp == null {TF, FT, FF}

amt < 0	**temp == null**	**Infeasible TRs**
T	T	X
T	F	
F	T	
F	F	

withdraw method

6. temp == null
7. amt <= 0
8. amt > temp.getBalance() {TFF, FTF, FFT, FFF}

temp == null	amt <= 0	amt > temp.getBalance()	Infeasible TRs
T	T	T	X
T	T	F	X
T	F	T	X
T	F	F	
F	T	T	X
F	T	F	
F	F	T	
F	F	F	

display method

9. getAccounts()[i] == null {T, F}