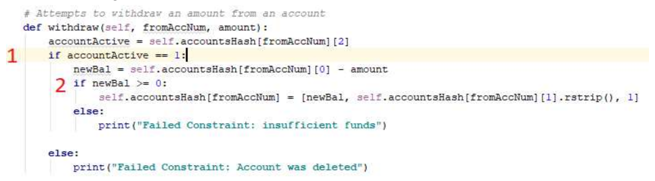
White Box Test #1: Withdraw

1. Code Section Being Tested:



```python
# Attempts to withdraw an amount from an account
def withdraw(self, fromAccNum, amount):
    accountActive = self.accountsHash[fromAccNum][2]
    if accountActive == 1:
        newBal = self.accountsHash[fromAccNum][0] - amount
        if newBal >= 0:
            self.accountsHash[fromAccNum] = [newBal, self.accountsHash[fromAccNum][1].rstrip(), 1]
        else:
            print("Failed Constraint: insufficient funds")

    else:
        print("Failed Constraint: Account was deleted")
```

2. Test Case Analysis

For the withdraw method, the team has decided to use decision coverage. Since there are two if statements (marked above with 1 and 2), there will be 4 possible test cases. (True-True, True-False, False-True, False-False). These cases will cover all possible decisions in this method.

3. Test Inputs

Test	Decision 1 (accountActive == 1)	Decision 2 (newBal > 0)	TSF statement
T1	1: true	1: true	WDR 1234567 1000 0000000 ***
T2	1: false	1: true	WDR 2468246 1000 0000000 ***
T3	2: true	2: false	WDR 1234567 999999999 0000000 ***
T4	2: false	2: false	WDR 2468246 999999999 0000000 ***

4. Test Results

The following test results were observed from the testing inputs above.

Test	Expected Result	Actual Result	Success (Y/N)	Explanation
T1	Successful transaction, 1000 withdrawn from account 1234567.	Same as expected	Y	No changes necessary
T2	Unsuccessful transaction, "Failed Constraint: Account was deleted"	ERROR	N	Small change in hash lookup to avoid crashing
T3	Unsuccessful transaction, "Failed Constraint: Account was deleted"	Same as expected	Y	No changes necessary
T4	Unsuccessful transaction, "Failed Constraint: insufficient funds"	Same as expected	Y	No changes necessary
	Unsuccessful transaction, "Failed Constraint: Account was deleted"	Same as expected	Y	No changes necessary

*Note: The testing was done manually, where the TSF statement was manually entered and the results were compared to the expected results by inspection.