CS 567 – Final Project Report

Abishek Pasupulate
ap3727@nau.edu
Northern Arizona University
Flagstaff, AZ-86001

Introduction:

The provided code is an implementation of a basic banking system in C++. It includes functionalities such as creating accounts, performing transactions (deposit, withdrawal, transfer), calculating interest, displaying account details, listing all accounts, searching accounts by owner name, and deleting accounts.

Code Source:

Code Structure:

The code consists of the following main components:

Banking System Class:

- Manages the banking system operations.
- Includes functions for creating accounts, performing transactions, calculating interest, displaying account details, and more.

Account Struct:

- Represents a bank account with attributes such as account number, owner name, balance, and transaction history.

Main Function:

- Demonstrates the usage of the BankingSystem class by creating accounts, performing transactions, and displaying account information.

Key Functions:

createAccount: Creates a new account with the specified account number, owner name, and initial balance.

deposit: Deposits a specified amount into the account with the given account number.

withdraw: Withdraws a specified amount from the account with the given account number.

transfer: Transfers a specified amount from one account to another.

calculateInterest: Calculates and applies interest to all accounts based on the given rate.

displayAccountDetails: Displays detailed information about a specific account.

displayAllAccounts: Displays information about all accounts in the banking system.

searchAccountsByOwner: Searches for accounts owned by a specific owner name and displays their details.

deleteAccount: Deletes the account with the specified account number.

Test File

The testing was performed using the **DeepState testing framework.**

Test Cases:

Account Creation:

- Verifies that accounts can be successfully created with valid input parameters.
- Randomly generates account number, owner name, and initial balance.
- Checks if the created account exists and has the correct attributes.

Deposit:

- Tests the deposit functionality by depositing a random amount into a randomly created account.
- Verifies that the account balance is updated correctly after the deposit.

Withdrawal:

- Tests the withdrawal functionality by withdrawing a random amount from a randomly created account.
- Verifies that the account balance is updated correctly after the withdrawal.

Transfer:

- Tests the transfer functionality by transferring a random amount from one randomly created account to another.
- Verifies that both source and destination account balances are updated correctly after the transfer.

Invalid Withdrawal:

- Checks the system's behavior when attempting to withdraw an invalid amount from an account.
- Verifies that the account remains unchanged and does not exist due to the invalid withdrawal.

Invalid Transfer:

- Checks the system's behavior when attempting to transfer an invalid amount from one account to another.
- Verifies that the source account remains unchanged and does not exist due to the invalid transfer.

Account Deletion:

- Tests the account deletion functionality by deleting a randomly created account.
- Verifies that the deleted account no longer exists in the system.

Pre-Testing:

- Pulled image and build the container
- Copied all the files to the container in /home/user/deepstate

```
PS C:\Users\ap3727\Desktop\ASA Project> docker cp 'C:\Users\ap3727\Desktop\ASA Project' 138a68304dc1fa2073703cf3137cf35445a06205705bf8141b98e12235160cbd:/home/user/deepst
ate/
Successfully copied 2.52MB to 138a68304dc1fa2073703cf3137cf35445a06205705bf8141b98e12235160cbd:/home/user/deepstate/
PS C:\Users\ap3727\Desktop\ASA Project> docker  exec -it 138a68304dc1fa2073703cf3137cf35445a06205705bf8141b98e12235160cbd /bin/bash
user@138a68304dc1:~/deepstate$
```

Basic Test:

Ran a Basic Test using **./test** and passed successfully

```
user@138a68304dc1:~/deepstate$ cd 'ASA Project'/
user@138a68304dc1:~/deepstate/ASA Project$ sudo clang++ banking_system.cpp test.cpp -o test -ldeepstate
user@138a68304dc1:~/deepstate/ASA Project$ ./test
Account created successfully.
Account created successfully.
Deposit successful. New balance: 6000
Withdrawal successful. New balance: 2500
Transfer successful. New balance for Alice: 5800
New balance for Bob: 2700
Account Number: 1001
Owner: Alice
Balance: 5800
Transaction History:
Transaction history for account 1001 (Alice):
Type: Deposit, Amount: 1000
Type: Transfer (to), Amount: 200
Account Number: 1002
Owner: Bob
Balance: 2700
Transaction History:
Transaction history for account 1002 (Bob):
Type: Withdrawal, Amount: 500
Type: Transfer (from), Amount: 200
List of all accounts:
Account Number: 1001, Owner: Alice, Balance: 5800
Account Number: 1002, Owner: Bob, Balance: 2700
Accounts owned by Alice:
Account Number: 1001, Balance: 5800
Account 1002 deleted successfully.
user@138a68304dc1:~/deepstate/ASA Project$ exit
```

Fuzzy Testing:

Fuzzy Test Has been Done by the following cmd

```
user@138a68304dc1:~/deepstate/ASA Project$ ./test --fuzz --output_test_dir d --timeout 30
```

This will generate random tests and Following is the output

```
Account created successfully.
Account created successfully.
Account created successfully.
Account created successfully.
Account created successfully.
INFO: Done fuzzing! Ran 65209 tests (2173 tests/second) with 0 failed/65209 passed/0 abandoned tests
user@138a68304dc1:~/deepstate/ASA Project$
```

Total 65209 tests has been ran in 30 secs

All Tests are passed and zero failed tests

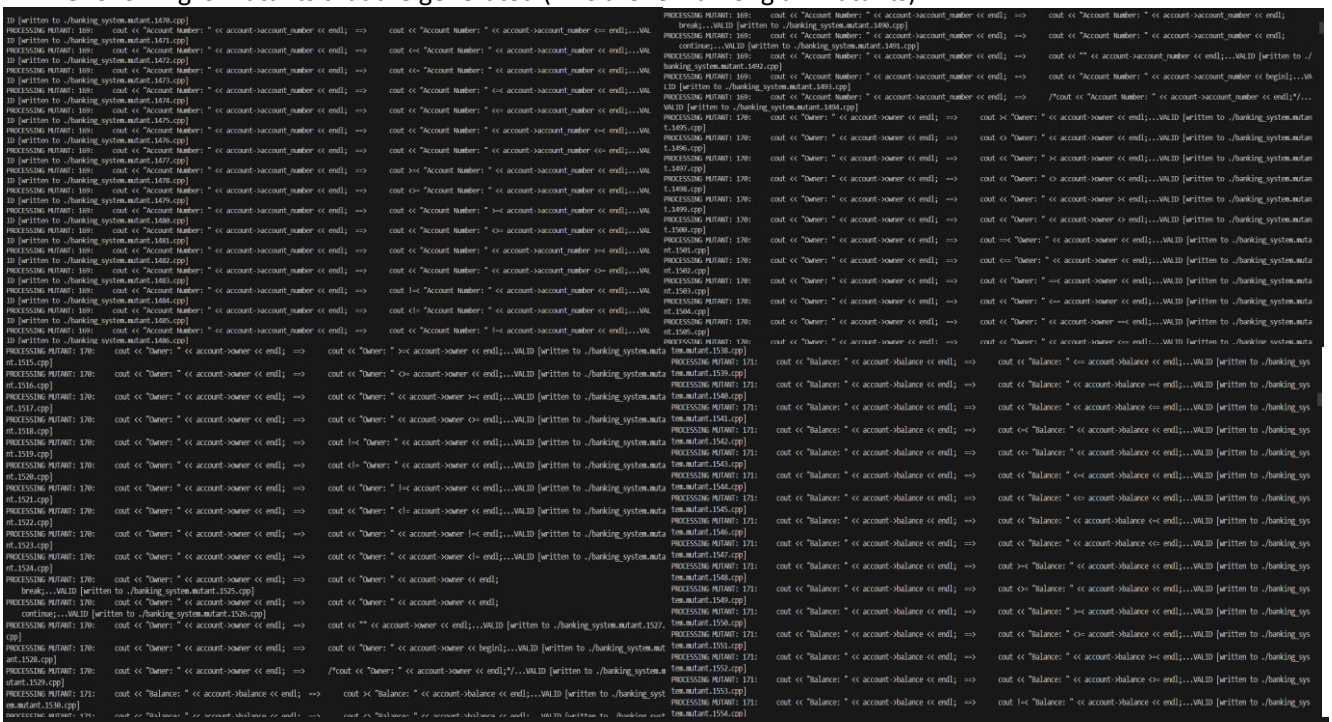
Universal Mutators:

Generating Mutants:

Mutants has been generated using the following cmd

```
user@138a68304dc1:~/deepstate/ASA Project$ sudo mutate banking_system.cpp
```

The following is mutants that are generated.(This are few among all mutants)



```
PROCESSING MUTANT: 222:     bank.deleteAccount(1002);  ==>      bank.deleteAccount(1);...VALID [written to ./banking_system.mutant.2023.cpp]
PROCESSING MUTANT: 222:     bank.deleteAccount(1002);  ==>      bank.deleteAccount(-1);...VALID [written to ./banking_system.mutant.2024.cpp]
PROCESSING MUTANT: 222:     bank.deleteAccount(1002);  ==>      bank.deleteAccount((1002+1));...VALID [written to ./banking_system.mutant.2025.cpp]
PROCESSING MUTANT: 222:     bank.deleteAccount(1002);  ==>      bank.deleteAccount((1002-1));...VALID [written to ./banking_system.mutant.2026.cpp]
PROCESSING MUTANT: 222:     bank.deleteAccount(1002);  ==>      bank.deleteAccount(1002);
    break;...VALID [written to ./banking_system.mutant.2027.cpp]
PROCESSING MUTANT: 222:     bank.deleteAccount(1002);  ==>      bank.deleteAccount(1002);
    continue;...VALID [written to ./banking_system.mutant.2028.cpp]
PROCESSING MUTANT: 222:     bank.deleteAccount(1002);  ==>      /*bank.deleteAccount(1002);*/...VALID [written to ./banking_system.mutant.2029.cpp]
PROCESSING MUTANT: 224:     return 0;  ==>      return 1;...VALID [written to ./banking_system.mutant.2030.cpp]
PROCESSING MUTANT: 224:     return 0;  ==>      return -1;...VALID [written to ./banking_system.mutant.2031.cpp]
PROCESSING MUTANT: 224:     return 0;  ==>      return (0+1);...VALID [written to ./banking_system.mutant.2032.cpp]
PROCESSING MUTANT: 224:     return 0;  ==>      return (0-1);...VALID [written to ./banking_system.mutant.2033.cpp]
PROCESSING MUTANT: 224:     return 0;  ==>      return 0;
    break;...VALID [written to ./banking_system.mutant.2034.cpp]
PROCESSING MUTANT: 224:     return 0;  ==>      return 0;
    continue;...VALID [written to ./banking_system.mutant.2035.cpp]
PROCESSING MUTANT: 224:     return 0;  ==>      /*return 0;*/...VALID [written to ./banking_system.mutant.2036.cpp]
PROCESSING MUTANT: 225: }*  ==>  }+/...VALID [written to ./banking_system.mutant.2037.cpp]
PROCESSING MUTANT: 225: }*  ==>  }-/...VALID [written to ./banking_system.mutant.2038.cpp]
PROCESSING MUTANT: 225: }*  ==>  }//...VALID [written to ./banking_system.mutant.2039.cpp]
PROCESSING MUTANT: 225: }*  ==>  }%/...VALID [written to ./banking_system.mutant.2040.cpp]
2041 VALID MUTANTS
0 INVALID MUTANTS
0 REDUNDANT MUTANTS
Valid Percentage: 100.0%

*****************************************************************************
WARNING:  because the handler does not compile and so has no pruning support,
all mutants were considered valid.  Consider using --cmd to build the target!
*****************************************************************************

user@138a68304dc1:~/deepstate/ASA Project$
```

Analyze the Mutants:

It Can be done using the following cmd

```
user@138a68304dc1:~/deepstate/ASA Project$ sudo analyze_mutants banking_system.cpp Project
```

Analyzation has been run for approx. 3 mins and these are few screenshots.

```
./banking_system.mutant.1870.cpp KILLED IN 0.053673744201660156 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1850: [109.57s 90.59% DONE]
RUNNING ./banking_system.mutant.1195.cpp...
./banking_system.mutant.1195.cpp KILLED IN 0.053290367126464844 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1851: [109.63s 90.64% DONE]
RUNNING ./banking_system.mutant.482.cpp...
./banking_system.mutant.482.cpp KILLED IN 0.05804443359375 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1852: [109.69s 90.69% DONE]
RUNNING ./banking_system.mutant.1145.cpp...
./banking_system.mutant.1145.cpp KILLED IN 0.0539507865905761 7 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1853: [109.74s 90.74% DONE]
RUNNING ./banking_system.mutant.468.cpp...
./banking_system.mutant.468.cpp KILLED IN 0.05757713317871094 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1854: [109.8s 90.79% DONE]
RUNNING ./banking_system.mutant.713.cpp...
./banking_system.mutant.713.cpp KILLED IN 0.05299544334411621 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1855: [109.86s 90.84% DONE]
RUNNING ./banking_system.mutant.448.cpp...
./banking_system.mutant.448.cpp KILLED IN 0.05895113945007324 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
```

```
RUNNING ./banking_system.mutant.635.cpp...
./banking_system.mutant.635.cpp KILLED IN 0.05470585823059082 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1881: [111.38s 92.11% DONE]
RUNNING ./banking_system.mutant.1867.cpp...
./banking_system.mutant.1867.cpp KILLED IN 0.05906081199645996 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1882: [111.44s 92.16% DONE]
RUNNING ./banking_system.mutant.462.cpp...
./banking_system.mutant.462.cpp KILLED IN 0.05561947822570801 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1883: [111.5s 92.21% DONE]
RUNNING ./banking_system.mutant.573.cpp...
./banking_system.mutant.573.cpp KILLED IN 0.055512189865112305 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1884: [111.56s 92.26% DONE]
RUNNING ./banking_system.mutant.748.cpp...
./banking_system.mutant.748.cpp KILLED IN 0.053359270095825195 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1885: [111.62s 92.31% DONE]
RUNNING ./banking_system.mutant.257.cpp...
./banking_system.mutant.257.cpp KILLED IN 0.05515146255493164 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1886: [111.68s 92.36% DONE]
RUNNING ./banking_system.mutant.761.cpp...
./banking_system.mutant.761.cpp KILLED IN 0.05363893508911133 (RETURN CODE 127)
  RUNNING SCORE: 1.0
```

```
#1899: [112.44s 92.99% DONE]
RUNNING ./banking_system.mutant.441.cpp...
./banking_system.mutant.441.cpp KILLED IN 0.060755252838134766 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1900: [112.51s 93.04% DONE]
RUNNING ./banking_system.mutant.1231.cpp...
./banking_system.mutant.1231.cpp KILLED IN 0.053960323333740234 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1901: [112.56s 93.09% DONE]
RUNNING ./banking_system.mutant.1078.cpp...
./banking_system.mutant.1078.cpp KILLED IN 0.05604100227355957 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1902: [112.62s 93.14% DONE]
RUNNING ./banking_system.mutant.1935.cpp...
./banking_system.mutant.1935.cpp KILLED IN 0.053343772888183594 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1903: [112.68s 93.19% DONE]
RUNNING ./banking_system.mutant.1014.cpp...
./banking_system.mutant.1014.cpp KILLED IN 0.055304765701293945 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1904: [112.74s 93.24% DONE]
RUNNING ./banking_system.mutant.1744.cpp...
./banking_system.mutant.1744.cpp KILLED IN 0.0599675178527832 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1905: [112.8s 93.29% DONE]
RUNNING ./banking_system.mutant.496.cpp...
./banking_system.mutant.496.cpp KILLED IN 0.05357623100280762 (RETURN CODE 127)
```

```
================================================================================
#1836: [108.74s 89.91% DONE]
RUNNING ./banking_system.mutant.1461.cpp...
./banking_system.mutant.1461.cpp KILLED IN 0.05654168128967285 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1837: [108.8s 89.96% DONE]
RUNNING ./banking_system.mutant.235.cpp...
./banking_system.mutant.235.cpp KILLED IN 0.05405735969543457 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1838: [108.86s 90.0% DONE]
RUNNING ./banking_system.mutant.1807.cpp...
./banking_system.mutant.1807.cpp KILLED IN 0.05842232704162598 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1839: [108.92s 90.05% DONE]
RUNNING ./banking_system.mutant.2025.cpp...
./banking_system.mutant.2025.cpp KILLED IN 0.053732872009277344 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1840: [108.98s 90.1% DONE]
RUNNING ./banking_system.mutant.78.cpp...
./banking_system.mutant.78.cpp KILLED IN 0.05781817436218262 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1841: [109.04s 90.15% DONE]
RUNNING ./banking_system.mutant.615.cpp...
./banking_system.mutant.615.cpp KILLED IN 0.054421186447143555 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#1842: [109.1s 90.2% DONE]
RUNNING ./banking_system.mutant.155.cpp...
```

The following screenshot is Overall Mutation Report.

The Mutation score is 1.

```
================================================================================
#2036: [120.74s 99.71% DONE]
RUNNING ./banking_system.mutant.296.cpp...
./banking_system.mutant.296.cpp KILLED IN 0.056687116622924805 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#2037: [120.81s 99.76% DONE]
RUNNING ./banking_system.mutant.788.cpp...
./banking_system.mutant.788.cpp KILLED IN 0.05901527404785156 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#2038: [120.87s 99.8% DONE]
RUNNING ./banking_system.mutant.1055.cpp...
./banking_system.mutant.1055.cpp KILLED IN 0.05332827568054199 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#2039: [120.93s 99.85% DONE]
RUNNING ./banking_system.mutant.1750.cpp...
./banking_system.mutant.1750.cpp KILLED IN 0.05512070655822754 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#2040: [120.98s 99.9% DONE]
RUNNING ./banking_system.mutant.698.cpp...
./banking_system.mutant.698.cpp KILLED IN 0.059259653091430664 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
#2041: [121.05s 99.95% DONE]
RUNNING ./banking_system.mutant.973.cpp...
./banking_system.mutant.973.cpp KILLED IN 0.05945587158203125 (RETURN CODE 127)
  RUNNING SCORE: 1.0
================================================================================
MUTATION SCORE: 1.0
user@138a68304dc1:~/deepstate/ASA Project$
```

Code Coverage:

Main function is added back to the code and reuploaded to the container again to check code coverage.

Code is compiled with main function using the following command

```
user@138a68304dc1:~/deepstate$ cd 'ASA Project'/
user@138a68304dc1:~/deepstate/ASA Project$ sudo clang++ --coverage -o coverage_op banking_system.cpp
```

Now we will run the out put file i.e. “**./coverage_op**”

```
user@138a68304dc1:~/deepstate/ASA Project$ sudo ./coverage_op
Account created successfully.
Account created successfully.
Deposit successful. New balance: 6000
Withdrawal successful. New balance: 2500
Transfer successful. New balance for Alice: 5800
New balance for Bob: 2700
Account Number: 1001
Owner: Alice
Balance: 5800
Transaction History:
Transaction history for account 1001 (Alice):
Type: Deposit, Amount: 1000
Type: Transfer (to), Amount: 200
Account Number: 1002
Owner: Bob
Balance: 2700
Transaction History:
Transaction history for account 1002 (Bob):
Type: Withdrawal, Amount: 500
Type: Transfer (from), Amount: 200
List of all accounts:
Account Number: 1001, Owner: Alice, Balance: 5800
Account Number: 1002, Owner: Bob, Balance: 2700
Accounts owned by Alice:
Account Number: 1001, Balance: 5800
Account 1002 deleted successfully.
user@138a68304dc1:~/deepstate/ASA Project$
```

Ln 197, Col 3 (2 se

Now we will run the command to get the code coverage.

```
user@138a68304dc1:~/deepstate/ASA Project$ sudo llvm-cov gcov banking_system.cpp
File './banking_system.h'
Lines executed:100.00% of 3
./banking_system.h:creating 'banking_system.h.gcov'

File '/usr/bin/../lib/gcc/x86_64-linux-gnu/7.5.0/../../../../include/c++/7.5.0/bits/basic_string.h'
Lines executed:100.00% of 4
/usr/bin/../lib/gcc/x86_64-linux-gnu/7.5.0/../../../../include/c++/7.5.0/bits/basic_string.h:creating 'basic_string.h.gcov'

File '/usr/bin/../lib/gcc/x86_64-linux-gnu/7.5.0/../../../../include/c++/7.5.0/bits/char_traits.h'
Lines executed:80.00% of 5
/usr/bin/../lib/gcc/x86_64-linux-gnu/7.5.0/../../../../include/c++/7.5.0/bits/char_traits.h:creating 'char_traits.h.gcov'

File '/usr/bin/../lib/gcc/x86_64-linux-gnu/7.5.0/../../../../include/c++/7.5.0/iostream'
Lines executed:100.00% of 1
/usr/bin/../lib/gcc/x86_64-linux-gnu/7.5.0/../../../../include/c++/7.5.0/iostream:creating 'iostream.gcov'

File 'banking_system.cpp'
Lines executed:80.14% of 141
banking_system.cpp:creating 'banking_system.cpp.gcov'

user@138a68304dc1:~/deepstate/ASA Project$
```

Ln 197, Col 3 (2 selected) Spaces: 4 UTF

My Final Code coverage is **80.14%**

Note: I have ran it multiple times, and every time it is same 80.14%