```
#!/bin/bash

#
# TEST SCRIPT FOR QIES
# CISC327 GROUP 13
# SPICE TESTS
# 01/11/2018
#
# to run:
#   You must have java installed on your linux system
#     run java -h to check if it is installed
#
#   QIES can be built, or you can use the provided .class files.
#     To build QUIES navigate to ./src/ directory and execute:
#     javac QIESBase.java -d ../bin/
#
#   To run QUIES navigate to ./build/ directory and execute:
#     java -cp ../bin "QIESBase" "vsf.txt"
#
#   Once java is installed, and QIES is built...
#     you may simply run ./run_tests.sh
#     and the tests will automatically be run using the
#     class files found in the /build/ directory.
#     The outputs of the console and the TXN summary
#     file from the ./output/ and ./expected/
#     are used for comparison.
#
#   On SUCCESS
#     Under the test name header SUCCESS will be displayed
#     This means console and txn summary files are matching
#
#   On FAIL
#     A detailed report of the console output as well as
#     differences between the txnsummary and console log
#     are displayed.
#
#   On COMPLETE
#     Counters of tests passed and tests failed are
#     displayed
#

# Counters
fails=0
successes=0
tests_run=0

# Clear past run
rm -rf output/*

# Parse inputs
for line in $(find . -iname 'input.txt'); do
  # Lines from input file
  value="$(cat $line)"

  # Test Directory and Name for
  # string building directories
  TESTDIR=$(dirname "${line}")
  CATDIR=$(dirname "${TESTDIR}")
  testname=${TESTDIR##*/}
  catname=${CATDIR##*/}

  # Make Output Directory
  mkdir -p "./output/${catname}/"
  mkdir -p "./output/${catname}/${testname}/"

  # Test header
```

```
  echo -e "\e[93m=========================\n${testname}\n=========================\e[0m"

  # Clear last txn summary
  rm "../build/transactions/txnsum.txt" 2> /dev/null

  # Run Test
  cp "./input/${catname}/${testname}/vsf.txt" "../build/vsf.txt"
  cd "../build/"
  run_output=$(echo -e "${value}\nexit" | java -cp ../bin "QIESBase" "vsf.txt")

  # Copy txn summary
  cd "../tests/"
  cp "../build/transactions/txnsum.txt" "./output/${catname}/${testname}/txnsum.txt"

  # Log console output
  echo -e "$run_output" > "output/${catname}/${testname}/console.log"

  # Compare files
  if cmp -s "expected/${catname}/${testname}/txnsum.txt"
  "output/${catname}/${testname}/txnsum.txt" &&
     cmp -s "expected/${catname}/${testname}/console.log"
     "output/${catname}/${testname}/console.log" ; then
    echo -e "\e[32mSUCCESS\e[0m\n"
    successes=$((successes+1))
  else
    echo -e "\e[31mFAILURE\e[0m\n"

    # Run console output
    echo -e "console output\n--------"
    echo -e "$run_output"

    # Txn summary differences
    echo -e "\ntxn summary diff\n--------"
    txn_diff_output=$(diff -y "expected/${catname}/${testname}/txnsum.txt"
    "output/${catname}/${testname}/txnsum.txt")
    echo -e "$txn_diff_output"
    echo -e "$txn_diff_output" > "output/${catname}/${testname}/txn_diff.log"

    # Console output differences
    echo -e "\nconsole output diff\n--------"
    con_diff_output=$(diff -y "expected/${catname}/${testname}/console.log"
    "output/${catname}/${testname}/console.log")
    echo -e "$con_diff_output"
    echo -e "$con_diff_output" > "output/${catname}/${testname}/con_diff.log"

    fails=$((fails+1))
  fi

  # Formatting
  echo -e "\n"

  # Increment total runs
  tests_run=$((tests_run+1))
done

# Output
echo -e "\e[104mTESTS COMPLETE:\e[0m\n  \e[31mFails:${fails}\n  \e[32mSuccesses:${successes}\n
\e[0mTests Run:${tests_run}\n"
```