Test Plan of SingerBallot

Bingan Feng/Isabella Wang

Negative:

The purpose of negative test cases are to test all the “modifiers” and “require” clauses, so as to verify its capability to reject all illegal operations.

Testing the “modifiers”:

1. validPhase():
   This modifier is to make sure specific operations must be executed in proper state. It is to be verified in 4 sub-cases (functions): register(), preVote(), finVote(), finalSingersRanking(). If any phase requirement is not met, the following error info would be thrown:

   ```
   [vm] from:0xca3...a733c to:Ballot.register(address) 0xbbf...732db value:0 wei data:0x442...92148 logs:0 hash:0xa83...ae4f2


   transact to Ballot.register errored: VM error: revert.
   revert  The transaction has been reverted to the initial state.
   Reason provided by the contract: "Invalid Phase.".      Debug the transaction to get more information.
   ```

2. onlyChair():
   This modifier is to make sure some specific operations can be performed by only the chairperson. It is to be verified in 4 sub-cases (functions): changeState(), markFinished(), register(). If any one is carried out by voters, the following error info would be thrown:

   ```
   [vm] from:0x147...c160c to:Ballot.changeState(uint8) 0xbbf...732db value:0 wei data:0x268...00002 logs:0 hash:0xd2b...bc86f


   transact to Ballot.changeState errored: VM error: revert.
   revert  The transaction has been reverted to the initial state.
   Reason provided by the contract: "Only Chairperson Has Authorization.". Debug the transaction to get more information.
   ```

3. notChair():
   This modifier is to make sure the chairperson doesn’t involve in the voting. It is to be verified in preVote() and finVote(). If the chairperson tries to vote, the following error info would be thrown:

   ```
   [vm] from:0xca3...a733c to:Ballot.finVote(uint256) 0xbbf...732db value:0 wei data:0xd2c...00000 logs:0 hash:0xd2f...e8412


   transact to Ballot.finVote errored: VM error: revert.
   revert  The transaction has been reverted to the initial state.
   Reason provided by the contract: "Chairperson Do Not Vote.".    Debug the transaction to get more information.
   ```

Testing the “require” clauses:

1. changeState:
   This ensures that the chairperson should only change to a later state, not changing back. If the chairperson tries to change to a previous phase, the following error info would be thrown:

```
[vm] from:0xca3...a733c to:Ballot.changeState(uint8) 0xbbf...732db value:0 wei data:0x268...00001 logs:0 hash:0x617...99f93


transact to Ballot.changeState errored: VM error: revert.
revert  The transaction has been reverted to the initial state.
Reason provided by the contract: "Cannot Go Back To The Previous State.".      Debug the transaction to get more information.
```

2. sender.prevoted > 0:
   This ensures that in pre-vote and final-vote phase, each voter has only 2 votes. If a voter tries to vote after using all the votes, the following error info would be thrown:

```
[vm] from:0x147...c160c to:Ballot.finVote(uint256) 0x987...04485 value:0 wei data:0xd2c...00002 logs:0 hash:0xcec...44415


transact to Ballot.finVote errored: VM error: revert.
revert  The transaction has been reverted to the initial state.
Reason provided by the contract: "All votes have been used.".   Debug the transaction to get more information.
```

3. valid singer index:
   This ensures that in pre-vote and final-vote phase, voters should only vote for a valid singer. For example, we have 3 singers (indexed as 0, 1, 2), if a voter tries to vote for “-1” or “5”, then the following error info would be thrown:

```
[vm] from:0x147...c160c to:Ballot.finVote(uint256) 0x987...04485 value:0 wei data:0xd2c...00003 logs:0 hash:0xaa0...dd7a8


transact to Ballot.finVote errored: VM error: revert.
revert  The transaction has been reverted to the initial state.
Reason provided by the contract: "Invalid Singer Index.".       Debug the transaction to get more information.
```

4. Cannot vote to the same singer:
   This ensures that in pre-vote and final-vote phase, voters cannot repeatedly vote for the same singer. If a voter tries to vote for singer 0 twice in pre-vote (or final-vote), the following error info would be thrown:

```
[vm] from:0x147...c160c to:Ballot.finVote(uint256) 0x987...04485 value:0 wei data:0xd2c...00000 logs:0 hash:0x942...7a129


transact to Ballot.finVote errored: VM error: revert.
revert  The transaction has been reverted to the initial state.
Reason provided by the contract: "Singer Already Been Voted.".  Debug the transaction to get more information.
```

5. Singer being voted should finish perform:
   This ensures that in pre-vote stage, voters can only voted for the singers that have already finished their performance. If a voter tries to vote for a singer (say singer 2) who is not finished yet, the following error info would be thrown:

```
[vm] from:0x147...c160c to:Ballot.preVote(uint256) 0xbbf...732db value:0 wei data:0x143...00002 logs:0 hash:0x684...f827c


transact to Ballot.preVote errored: VM error: revert.
revert  The transaction has been reverted to the initial state.
Reason provided by the contract: "Singer Not Finished.".        Debug the transaction to get more information.
```

Positive:

1. Use chairperson account to deploy 3 singers. The state is initialized to be Phase.Regs.

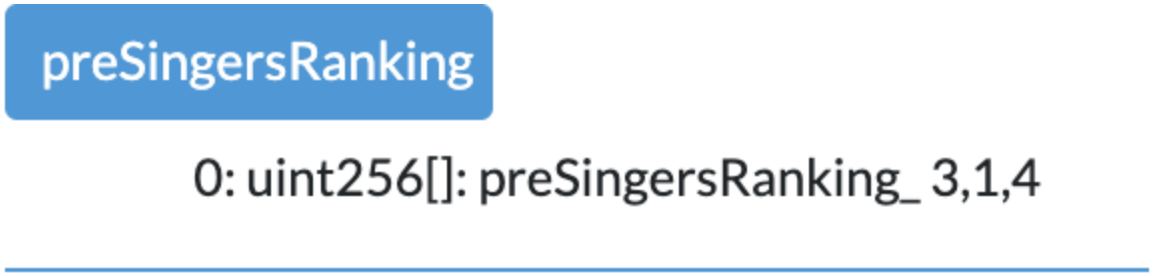
2. In Phase.Regs, use chairperson account to registers 4 voters with function register(address voter).

3. Use chairperson account to change state to Phase.Prevote with function changeState(Phase x).

4. Use chairperson to mark singer 0,1,2 as finished with function markFinished(uint singer)

5. Use account 0x147… to vote singer 0,2 with function preVote(uint singer)
   Use account 0x4B0… to vote singer 0,2 with function preVote(uint singer)
   Use account 0x583… to vote singer 0,2 with function preVote(uint singer)
   Use account 0xdD8… to vote singer 1,2 with function preVote(uint singer)

6. Use account 0x147... to call function preSingersRanking()



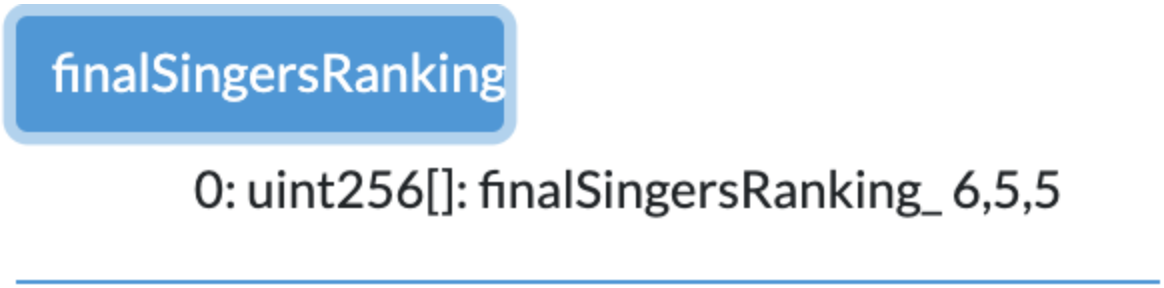
   3,1,4 represents singer0 gets 3 votes, singer1 gets 1 votes and singer 2 gets 2 votes, which is the same as expected. The ranking will be done by JavaScript in part 2.

7. Use chairperson account to change state to Phase.FinVote with function changeState(Phase x).

8. Use account 0x147… to vote singer 0,1 with function finVote(uint singer)
   Use account 0x4B0… to vote singer 0,1 with function finVote(uint singer)
   Use account 0x583… to vote singer 0,1 with function finVote(uint singer)
   Use account 0xdD8… to vote singer 1,2 with function finVote(uint singer)

9. Use chairperson account to change state to Phase.Done with function changeState(Phase x).

10. Use account 0x147... to call function finalSingersRanking()



6,5,5 represents totally singer0 gets 6 votes, singer1 gets 5 votes and singer 2 gets 5 votes, which is the same as expected. The ranking will be done by JavaScript in part 2.