| **Project Name: Project 1: Voting System Team# 9** | |
| --- | --- |
| **Test Stage: Unit** | **Test Date: 3/21/2021** |
| **Test Case ID#: ballotObjTest** | **Name(s) of Testers: Abdikarim Fareh** |
| **Test Description:**  **we are testing to ensure that each ballot**  **created has an id associated with. And can return**  **ballot array and it’s id correctly.**  **we also testing to make sure that we get the correct**  **round during distribution and who ever is what possiotion**  **during distribution (round/rank)** |  |
| **Automated: Yes** | **Indicate where you are storing the tests (what file) and the name of the method/functions being used.**  **This was tested by creating an array manual inside the BallotTest class and initialized and used the Ballot constructor to do the operation.** |
| **Results: Pass** |  |
|  |  |
| **Preconditions for Test :**  **given an array of Ballot and id.** | |

| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
| 1 | **#testGetIDBallot()**  upon a ballot is instantiated, it will create an array of ballot that is associated with an ID. we are testing to make sure that it returns the appropriate ID number. | // for IR  int id\_a1 = 1,  int id\_a2 = 2  // for OPL  itn id\_a3 = 1  int id\_a4 = 2 | Expected results for IDs : 1, and 2 for IR  Expected results for IDs : 1, and 2 for OPL | actual result are : 1 and 2  for IR  And  Actual result are : 1 and 2  for IR | For each ballot created, it is associated with an ID number for tracking purposes. we are making sure that it’s return the appropriate ID number |
| 2 | **testGetChoiceAsArray()**  For each ballot that represents an array is tested. Given an array, can we get the array choices exactly as expected | int a1 [] = {1,2,3,4,5}  int a3 [] = {2,3, 4,5,1}  int a2 [] = {0, 0,0,1,0}  int a3 [] = {0, 1,0,0,0} | we expected 2 results.  Result 1 = {1,2,3,4,5}  Result 2 = {2,3,4,5,1}  Result 3 = {0, 0,0,1,0}  Result 4 = {0, 1,0,0,0} | A\_Result 1 = {1,2,3,4,5}  A\_Result 2 = {2,3,4,5,1}  A\_Result 3 = {0, 0,0,1,0}  A\_Result 4 = {0, 1,0,0,0} | we tested different Ballot for IR and OPL to get the correct return for each Ballot that is created |
| 3 | **testGetIndexOfNthChoices()**  we testing Nth Choice for IR where we need to find who is what index so that we get Nth choice for redistribution for ballots | round\_a1 = {1,2,3,4,5}  round\_a2 = {2,3,4,5,1}  round\_a3 = {0,0,0,1,0}  round\_a4 = {0,1,0,0,0} | Given 2, the expected result should be returned equal to index 1.  Given 5, the expected result should be returned equal to index 3.  Given, 1 -> expected result is 3rd and 1st index respectively | The expected results are index 1 and 3 respectively.  For OPL, expected result are 3rd and 1. | this method returns nth choice for a given ballot. Let’s say, we are doing round 3 re-distribution. Given 3, can you return it’s index |
| 4 | **testRound()**  We also tested what the current round is for a given Ballot. This is to make sure that we are doing the distribution correctly. | round = 1 // initially    Initially all rounds are 1 | Expect result is 1 | actual result is 1 | we are testing to make sure that we get the correct round count. |
| 5 | **test2ndRound()**  after incrementing round, can we get the accurate result for next round | First -> increment) adds 1 to round  secondly, by increment by 1, all rounds should be 2 | Expt = 2 and for second round | actual result is 2 for all four test | we are testing to make sure that we get the correct round count. We also making sure that we increment round to next number to get an accurate results |

**Post condition(s) for Test :**

Ballot outputs the actual result that we expected after testing with different Ballots. 

**Test Passed : Yes**