|  |  |
| --- | --- |
| **Project Name: Project 1: CompuVote Team #19** | |
| **Test Stage:** Unit | **Test Date:** 3/27/21 |
| **Test Case ID #:** test\_394\_07\_getLowestHighestCandidates\_02\_testGetLowestHighestCandidatesMultipleLowest | **Name(s) of Testers:** Aaron Kandikatla, Jack Fornaro |
| **Test Description:**  Given a standard IR election with candidates, parties, and ballots, where there is one clear candidate with the highest votes and many candidates with the lowest votes, tests that getLowestHighestCandidates returns all the lowest candidates and one highest candidate |  |
| **Automated:** Yes | **Indicate where you are storing the tests (what file) and the name of the method/functions being used:**  Test file: Project1/src/test/org/team19/InstantRunoffSystemTest.java  Test method: testGetLowestHighestCandidatesMultipleLowest  Method/constructor being tested: getLowestHighestCandidates from Project1/src/main/org/team19/InstantRunoffSystem.java |
| **Results:** Pass |  |
|  |  |
| **Preconditions for Test:**  N/A | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
|  |  |  |  |  |  |
| 1 | Tests that all candidates with the lowest votes (0) Kleinberg, Loser, Bobster; are returned by getLowestHighestCandidates | Candidates:  Rosen (D) – 3 ballots  Kleinberg (R) – 0 ballots  Chou (I) – 2 ballots  Royce (L) – 1 ballot  Loser (L) – 0 ballots  Bobster (I) – 0 ballots | "Pair{Pair{0, [Kleinberg (R), Loser (L), Bobster (I)]}, Pair{3, Rosen (D)}}" | "Pair{Pair{0, [Kleinberg (R), Loser (L), Bobster (I)]}, Pair{3, Rosen (D)}}" | N/A |

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

N/A