|  |  |
| --- | --- |
| **Project Name: Aegis Project 1: Voting System Team#7** | |
| **Test Stage: Unit X System \_** | **Test Date:** 3/27/21 |
| **Test Case ID#:** Helper\_1 | **Name(s) of Testers:** Michael/Donald/Lucky/Grant |
| **Test Description:**  Tests that the string input taken for  an OPL ballot returns the correct index  if it’s supposed to. |  |
| **Automated: yes X no \_\_\_** | **Indicate where you are storing the tests (what file) and the name of the method/functions being used.**  Stored in file:  Method names:   * OPLBallotToIndex() |
| **Results: Pass \_\_X\_\_\_** Fail **\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  The strings used for the tests have been set up correctly. | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
|  |  |  |  |  |  |
| 1 | Test for random string | std::string garbage = "fjdkasl;"; | std::string::npos | std::string::npos |  |
| 2 | Test for empty string | std::string empty = ""; | std::string::npos | std::string::npos |  |
| 3 | Test for string with no zeroes | std::string no\_ones = ",,,,"; | std::string::npos | std::string::npos |  |
| 4 | Test for a 1 in the first position | std::string first\_pos = “1,,,,”; | 0 | 0 |  |

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

The value returned should be the correct index if the string is a valid string..