|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: promptFile\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_file\_name() can prompt a file properly.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/promptFileTest.cc**  **Name of the method:**  **get\_file\_name** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **User enter a file name.** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make promptFileTest  $./promptFileTest CPL.CSV | CPL.csv | CPL.csv | CPL.csv |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The correct file is found.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: generateAuditFile\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_type\_of\_election() can get proper election type of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ generateAuditFileTest.cc**  **Name of the method:**  **get\_type\_of\_election()** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **A file is successfully prompted.** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make generateAuditFileTest  $./generateAuditFileTest CPL.CSV | CPL.CSV | CPL | CPL |  |
| 2 | $Make generateAuditFileTest  $./generateAuditFileTest OPL.CSV | OPL.CSV | OPL | OPL |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The correct voting list type is found.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: CPL\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_type\_of\_election() can get proper election type of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ cplTest.cc**  **Name of the method:**  **get\_type\_of\_election()** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is CPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make cplTest  $./cplTest CPL.CSV | CPL.CSV | CPL | CPL |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The correct voting list type is found.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: OPL\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_type\_of\_election() can get proper election type of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ oplTest.cc**  **Name of the method:**  **get\_type\_of\_election()** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is OPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make oplTest  $./cplTest | OPL.CSV | OPL | OPL |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The correct voting list type is found.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: CPL\_02** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_number\_of\_parties() can get proper number of parties of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ cplTest.cc**  **Name of the method:**  **get\_number\_of\_parties** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is CPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make cplTest  $./cplTest | CPL.CSV | 4 | 4 |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The number of parties is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: CPL\_03** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_number\_of\_seat() can get proper number of seat of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ cplTest.cc**  **Name of the method:**  **get\_number\_of\_seat** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is CPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make cplTest  $./cplTest | CPL.CSV | 7 | 7 |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The number of seats is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: OPL\_03** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_number\_of\_seat() can get proper number of seat of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ oplTest.cc**  **Name of the method:**  **get\_number\_of\_seat** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is OPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make oplTest  $./oplTest | OPL.CSV | 3 | 3 |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The number of seats is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: CPL\_04** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_number\_of\_candidate() can get proper number of candidate of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ cplTest.cc**  **Name of the method:**  **get\_number\_of\_ candidate** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is CPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make cplTest  $./cplTest | CPL.CSV | 16 | 16 |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The number of candidate is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: OPL\_04** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_number\_of\_candidate() can get proper number of candidate of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ oplTest.cc**  **Name of the method:**  **get\_number\_of\_ candidate** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is OPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make oplTest  $./oplTest | OPL.CSV | 6 | 6 |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The number of candidate is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: CPL\_05** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_number\_of\_ballot() can get proper number of ballot of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ cplTest.cc**  **Name of the method:**  **get\_number\_of\_ ballot** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is CPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make cplTest  $./cplTest | CPL.CSV | 100 | 100 |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The number of ballot is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: OPL\_05** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_number\_of\_ballot() can get proper number of ballot of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ oplTest.cc**  **Name of the method:**  **get\_number\_of\_ ballot** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is OPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make oplTest  $./oplTest | OPL.CSV | 9 | 9 |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The number of ballot is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: CPL\_06** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_candidate() can get proper candidate of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ cplTest.cc**  **Name of the method:**  **get\_candidate** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is CPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make cplTest  $./cplTest | CPL.CSV | [Pike,0,1] | [Pike,0,1] |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The candidate output is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: OPL\_06** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_candidate() can get proper candidate of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ oplTest.cc**  **Name of the method:**  **get\_candidate** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is OPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make oplTest  $./oplTest | OPL.CSV | [Pike,D] | [Pike,D] |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The candidate output is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: CPL\_07** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_ballot\_ordering() can get proper ballot ordering of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ cplTest.cc**  **Name of the method:**  **get\_ ballot\_ordering** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is CPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make cplTest  $./cplTest | CPL.CSV | D | D |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The ballot ordering output is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: CPL\_08** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_CPL\_vote() can get proper CPL vote of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ cplTest.cc**  **Name of the method:**  **get\_CPL\_vote** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is CPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make cplTest  $./cplTest | CPL.CSV | 1,,, | 1,,, |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The CPL vote output is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: OPL\_08** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_OPL\_vote() can get proper CPL vote of the file.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ oplTest.cc**  **Name of the method:**  **get\_OPL\_vote** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The election type is OPL** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make oplTest  $./oplTest | OPL.CSV | 1,,,,, | 1,,,,, |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The OPL vote output is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: countVoteCPL\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_vote\_count() can get proper CPL vote** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ countVoteTest.cc**  **Name of the method:**  **get\_vote\_count** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The CPL audit file is created** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make countVoteTest  $./countVoteTest | CPL.CSV | D 3 | D 3 |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The CPL vote output is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: countVoteOPL\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the get\_vote\_count() can get proper CPL vote** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ countVoteOPLTest.cc**  **Name of the method:**  **get\_vote\_count** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **The OPL audit file is created** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make countVoteOPLTest  $./countVoteOPLTest | OPL.CSV | [Borg,R] | [Borg,R] |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The OPL vote output is correct.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: displayWinner\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the display() can random select a winner when there is a tie vote.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ displayWinnerTest.cc**  **Name of the method:**  **display** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **Audit file is generated** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make displayWinnerTest  $./displayWinnerTest | (“Jhon”,3)  (“Parker”,10)  (“James”,5) | The winner is parker  winner has total vote 10 | The winner is parker  winner has total vote 10 |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The correct winner is displayed and the total vote as well.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: displayWinner\_02** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the tie\_handler() can random select a winner when there is a tie vote.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ displayWinnerTest.cc**  **Name of the method:**  **tie\_handler** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **Audit file is generated** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make displayWinnerTest  $./displayWinnerTest |  | 0 | 0 |  |
| 2 | $Make displayWinnerTest  $./displayWinnerTest |  | 1 | 1 | Since it’s random output. Thus if we run twice ending with two different output, then it should be correct. |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

A winner is randomly selected when there is a tie vote.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: displayWinnerCPL\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the display() can random select a winner when there is a tie vote.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ displayWinnerCPLTest.cc**  **Name of the method:**  **display** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **Audit file is generated** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make displayWinnerCPLTest  $./displayWinnerCPLTes | (“A”,3)  (“B,10)  (“C”,5)  [John,A,1]  [Parker,B,1] | The winner party is B  with total vote 10  the party B has the following candidate(s):  [Parker,B,1] | The winner party is B  with total vote 10  the party B has the following candidate(s):  [Parker,B,1] |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

The correct winner party with its total vote and its candidates are displayed.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_x\_ System \_\_** | **Test Date: 11-17-2019** |
| **Test Case ID#: displayWinnerCPL\_02** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the tie\_handler() can random select a winner when there is a tie vote.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ displayWinnerCPLTest.cc**  **Name of the method:**  **tie\_handler** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **Audit file is generated** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make displayWinnerCPLTest  $./displayWinnerCPLTest |  | 0 | 0 |  |
| 2 | $Make displayWinnerTest  $./displayWinnerCPLTest |  | 1 | 1 | Since it’s random output. Thus if we run twice ending with two different output, then it should be correct. |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

A winner is randomly selected when there is a tie vote.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_\_ System \_x\_** | **Test Date: 11-18-2019** |
| **Test Case ID#: OPL\_System\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the OPL workflow works properly.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/** **OPLSystemTest.cc**  **Name of the method:**  **OPL** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **An OPL file is provided as input.** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make OPLSystemTest | OPL.CSV | AuditOPL.txt | AuditOPL.txt |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

Successfully generated an audit file according to OPL file.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_\_ System \_x\_** | **Test Date: 11-18-2019** |
| **Test Case ID#: CPL\_System\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the the CPL workflow works properly.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/CPLSystemTest.cc**  **Name of the method:**  **CPL** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **A CPL file is provided as input.** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make CPLSystemTest | CPL.CSV | AuditCPL.txt | AuditCPL.txt |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

Successfully generated an audit file according to CPL file.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_\_ System \_x\_** | **Test Date: 11-18-2019** |
| **Test Case ID#: CPL\_Tie\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the tie handler works properly under CPL.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/CPLSystemTest.cc**  **Name of the method:**  **CPL** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **A CPL file is provided as input and there is a tie.** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make CPLSystemTest  $./CPLSystemTest ../testing/CPL\_Tie.CSV | CPL\_Tie.CSV | The type of election is: CPL  The numer of parties is: 4  The numer of sets is: 7  The numer of ballots is: 100  The numer of candidates is: 16  the winner party is R with total votes 4  the party R has the following candidate(s):  [Deutsch,R,1]  [Wong,R,2]  [Walters,R,3]  [Keller,R,4]  [Borg,R,5] | The type of election is: CPL  The numer of parties is: 4  The numer of sets is: 7  The numer of ballots is: 100  The numer of candidates is: 16  the winner party is R with total votes 4  the party R has the following candidate(s):  [Deutsch,R,1]  [Wong,R,2]  [Walters,R,3]  [Keller,R,4]  [Borg,R,5] |  |
| 2 | $Make CPLSystemTest  $./CPLSystemTest ../testing/CPL\_Tie.CSV | CPL\_Tie.CSV | The type of election is: CPL  The numer of parties is: 4  The numer of sets is: 7  The numer of ballots is: 100  The numer of candidates is: 16  the winner party is G with total votes 4  the party G has the following candidate(s):  [Jones,G,1]  [Smith,G,2]  [Lewis,G,3]  [Smith,G,4]  [Li,G,5] | The type of election is: CPL  The numer of parties is: 4  The numer of sets is: 7  The numer of ballots is: 100  The numer of candidates is: 16  the winner party is G with total votes 4  the party G has the following candidate(s):  [Jones,G,1]  [Smith,G,2]  [Lewis,G,3]  [Smith,G,4]  [Li,G,5] | We tested the target file twice to should that tie handler will return random winner while there is a tie. |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

Winner is randomly generated while there is tie in CPL case.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_\_ System \_x\_** | **Test Date: 11-18-2019** |
| **Test Case ID#: OPL\_Tie\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests if the tie handler works properly under OPL.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/** **OPLSystemTest.cc**  **Name of the method:**  **OPL** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **An OPL file is provided as input and there is a tie.** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | Expected  Result | Actual  Result | **Notes** |
| 1 | $Make OPLSystemTest  $./OPLSystemTest ../testing/OPL\_Tie.CSV | OPL\_Tie.CSV | The type of election is: OPL  The numer of sets is: 3  The numer of ballots is: 9  The numer of candidates is: 6  the winner is [Foster,D]  the wiiner has total votes 4 | The type of election is: OPL  The numer of sets is: 3  The numer of ballots is: 9  The numer of candidates is: 6  the winner is [Foster,D]  the wiiner has total votes 4 |  |
| 2 | $Make OPLSystemTest  $./OPLSystemTest ../testing/OPL\_Tie.CSV | OPL\_Tie.CSV | The type of election is: OPL  The numer of sets is: 3  The numer of ballots is: 9  The numer of candidates is: 6  the winner is [Pike,D]  the wiiner has total votes 4 | The type of election is: OPL  The numer of sets is: 3  The numer of ballots is: 9  The numer of candidates is: 6  the winner is [Pike,D]  the wiiner has total votes 4 | We tested the target file twice to should that tie handler will return random winner while there is a tie. |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

Winner is randomly generated while there is tie in OPL case.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_\_ System \_x\_** | **Test Date: 11-18-2019** |
| **Test Case ID#: OPL\_Boundary\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests the efficiency of the code and run time.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/** **OPLSystemTest.cc**  **Name of the method:**  **OPL** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **An OPL file is provided as input with large data.** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make OPLSystemTest  $./OPLSystemTest ../testing/OPL\_Large.CSV | OPL\_Large.CSV | The type of election is: OPL  The numer of sets is: 3  The numer of ballots is: 26100  The numer of candidates is: 6  the winner is [Foster,D]  the wiiner has total votes 9565 | The type of election is: OPL  The numer of sets is: 3  The numer of ballots is: 26100  The numer of candidates is: 6  the winner is [Foster,D]  the wiiner has total votes 9565 | Run time is within the required time frame. |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

Run time is within the required time frame.

|  |  |
| --- | --- |
| **Project Name: Project 1: Voting System Team#26** | |
| **Test Stage: Unit \_\_ System \_x\_** | **Test Date: 11-18-2019** |
| **Test Case ID#: CPL\_Boundary\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests the efficiency of the code and run time.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ C****PLSystemTest.cc**  **Name of the method:**  **CPL** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **An CPL file is provided as input with large data.** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | $Make CPLSystemTest  $./CPLSystemTest ../testing/CPL\_Large.CSV | CPL\_Large.CSV | The type of election is: CPL  The numer of parties is: 4  The numer of sets is: 7  The numer of ballots is: 21242  The numer of candidates is: 16  the winner party is R with total votes 8187  the party R has the following candidate(s):  [Deutsch,R,1]  [Wong,R,2]  [Walters,R,3]  [Keller,R,4]  [Borg,R,5] | The type of election is: CPL  The numer of parties is: 4  The numer of sets is: 7  The numer of ballots is: 21242  The numer of candidates is: 16  the winner party is R with total votes 8187  the party R has the following candidate(s):  [Deutsch,R,1]  [Wong,R,2]  [Walters,R,3]  [Keller,R,4]  [Borg,R,5] | Run time is within the required time frame. |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| 4 |  |  |  |  |  |
|  |  |  |  |  |  |

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

Run time is within the required time frame.

|  |  |
| --- | --- |
| **Project Name: Voting System Team#26** | |
| **Test Stage: Unit \_\_ System \_x\_** | **Test Date: 12-10-2019** |
| **Test Case ID#: OPL\_Regression\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests the completion of OPL.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ Q7\_QFile.pro** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **An OPL file is provided as input.** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | 1. Type ctrl + r to run the program in QT Creator  2. Click read file button to select OPL.CSV file in testing folder  3. Click run to display the result of OPL voting  4. Click save file to save the generated audit file  5. Name the audit file OPLTest1.txt | OPL.CSV | OPLTest1.txt | OPLTest1.txt | Everything in OPLTest1.txt shows the correct results |
| 2 | 1. Type ctrl + r to run the program in QT Creator  2. Click read file button to select OPL\_Large.CSV file in testing folder  3. Click run to display the result of OPL voting  4. Click save file to save the generated audit file  5. Name the audit file OPLTest2.txt | OPL\_Large.CSV | OPLTest2.txt | OPLTest2.txt | Everything in OPLTest2.txt shows the correct results |
| 3 | 1. Type ctrl + r to run the program in QT Creator  2. Click read file button to select OPL\_Tie.CSV file in testing folder  3. Click run to display the result of OPL voting  4. Click save file to save the generated audit file  5. Name the audit file OPLTest3.txt | OPL\_Tie.CSV | OPLTest3.txt | OPLTest3.txt | Everything in OPLTest3.txt has correct outputs and shows the winner is [Foster,D] |
| 4 | 1. Type ctrl + r to run the program in QT Creator  2. Click read file button to select OPL\_Tie.CSV file in testing folder  3. Click run to display the result of OPL voting  4. Click save file to save the generated audit file  5. Name the audit file OPLTest4.txt | OPL\_Tie.CSV | OPLTest4.txt | OPLTest4.txt | Everything in OPLTest4.txt has correct outputs and shows the winner is [Pike,D]. which means tie handler works properly. |
|  |  |  |  |  |  |

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

The correct audit file is generated and saved.

|  |  |
| --- | --- |
| **Project Name: Voting System Team#26** | |
| **Test Stage: Unit \_\_ System \_x\_** | **Test Date: 12-10-2019** |
| **Test Case ID#: CPL\_Regression\_01** | **Name(s) of Testers:**  **Songyan Wu**  **Ge Yu**  **Hongyu Zhu** |
| **Test Description:**  **This tests the completion of CPL.** |  |
| **Automated: yes\_\_\_ no \_x\_\_** | **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**  **Test file path:**  **umn-csci5801-f19/repo-Team26/Project1/src/ Q7\_QFile.pro** |
| **Results: Pass \_\_x\_\_\_ Fail\_\_\_\_\_\_\_\_** |  |
|  |  |
| **Preconditions for Test:**  **An CPL file is provided as input.** | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step**  **#** | **Test Step**  **Description** | **Test**  **Data** | **Expected**  **Result** | **Actual**  **Result** | **Notes** |
| 1 | 1. Type ctrl + r to run the program in QT Creator  2. Click read file button to select CPL.CSV file in testing folder  3. Click run to display the result of CPL voting  4. Click save file to save the generated audit file  5. Name the audit file CPLTest1.txt | CPL.CSV | CPLTest1.txt | CPLTest1.txt | Everything in CPLTest1.txt shows the correct results |
| 2 | 1. Type ctrl + r to run the program in QT Creator  2. Click read file button to select CPL\_Large.CSV file in testing folder  3. Click run to display the result of CPL voting  4. Click save file to save the generated audit file  5. Name the audit file CPLTest2.txt | CPL\_Large.CSV | CPLTest2.txt | CPLTest2.txt | Everything in CPLTest2.txt shows the correct results |
| 3 | 1. Type ctrl + r to run the program in QT Creator  2. Click read file button to select CPL\_Tie.CSV file in testing folder  3. Click run to display the result of CPL voting  4. Click save file to save the generated audit file  5. Name the audit file CPLTest3.txt | CPL\_Tie.CSV | CPLTest3.txt | CPLTest3.txt | Everything in CPLTest3.txt has correct outputs and shows the winner party is R |
| 4 | 1. Type ctrl + r to run the program in QT Creator  2. Click read file button to select CPL\_Tie.CSV file in testing folder  3. Click run to display the result of CPL voting  4. Click save file to save the generated audit file  5. Name the audit file CPLTest4.txt | CPL\_Tie.CSV | CPLTest4.txt | CPLTest4.txt | Everything in CPLTest4.txt has correct outputs and shows the winner party is G, which means tie handler works properly. |
|  |  |  |  |  |  |

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

The correct audit file is generated and saved.

|  |
| --- |
| Test 4: OPL System test |
| Team Members Responsible:  Hongyu Zhu, Songyan Wu, Ge Yu  Inputs:  1.  Tests:  Outputs:  Passed or Failed:  Date:12-10-2019 |