Project Name: Project 1 Voting System

Team #9

Test Stage: Unit __ System X	**Test Date: 5/1/22**
Test Case ID#: OPL_System_1	**Name(s) of Testers: John Cullom**
Test Description: Test to see if OPL algorithm can reallocate seats correctly if a party receives more seats than candidates	
	Indicate where are you storing the tests (what file) and the name of the method/functions being used. **Input_OPL.csv (run according to ReadMe instructions)**
Automated: yes___ no X	

Results: Pass X Fail

Preconditions for Test: Input file must be in the folder as the system's java files so that system can access it.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Place the Inout_OPL.csv file in testing/InputFiles from the testing folder. Navigate to the Project1 folder and type in “javac -d . src/*.java” And then type in “java p1.Main testing/InputFiles”	Input_OPL.csv			
2	Verify that result.file has the correct election results	resultFile.txt	Seats are given to Pike,Deutsch, Foster. And Party D received 7 votes Party R received 4 votes	Seats are given to Pike, Deutsch, Foster. And Party D received 7 votes Party R received 4 votes Party I received 2 votes	

			Party I received 2 votes		
3	Verify that the election process in audit.file	auditFile.txt	The election process can be followed and there are no noticeable errors. Pike receives a vote through achieving droop quota and Deutsch and Foster receive seats through remainders	The election process can be followed and there are no noticeable errors. Pike receives a vote through achieving droop quota and Deutsch and Foster receive seats through remainders	
4					
5					

Post condition(s) for Test: New audit and result files are generated and system successfully finishes running.

Project Name: Project 1 Voting System

Team #9

Test Stage: Unit __ System X **Test Date: 4/18/22**

Test Case ID#: OPL_System_2 **Name(s) of Testers: Anna Frenz**

Test Description: Test to see if OPL algorithm can reallocate seats correctly if a party receives more seats than candidates

Indicate where are you storing the tests (what file) and the name of the method/functions being used.

Input_OPL1.csv (run according to ReadMe instructions)

Automated: yes___ no X

Results: Pass X Fail

Preconditions for Test: Input file must be in the folder as the system's java files so that system can access it.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes

1	Place the Inout_OPL1.csv file in testing/InputFiles from the testing folder. Navigate to the Project1 folder and type in “javac -d . src/*.java” And then type in “java p1.Main testing/InputFiles”	Input_OPL1.csv			
2	Verify that the election process in auditFile.txt	Verify that the election process in auditFile.txt	The election process can be followed and there are no noticeable errors. 2nd independent seat is redistributed to R party	Each party’s top candidate receives one seat as expected.	
3					
4					
5					

Post condition(s) for Test: New audit and result files are generated and system successfully finishes running.

Project Name: Project 1 Voting System

Team #9

Test Stage: Unit __ System X	**Test Date: 4/19/22**
Test Case ID#: OPL_System_3	**Name(s) of Testers: Naviin Vejaya Kumar**
Test Description: Test to see if OPL algorithm can reallocate seats correctly if a party receives more seats than candidates	
	Indicate where are you storing the tests (what file) and the name of the method/functions being used. **Input_OPL4.csv (run according to ReadMe instructions)**
Automated: yes___ no X	

Results: Pass X Fail

Preconditions for Test: Input file must be in the folder as the system's java files so that system can access it.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Navigate to the Project1 folder and type in “javac -d . src/*.java testing/*.java” And then type in “java p1.Main src/Input_OPL4.csv”	Input_OPL4.csv			
2	Verify that the election process in auditFile.txt	auditFile.txt	The Democratic party should be allocated 2 seats by meeting quota and one by remainder, should be able to follow election logic	The Democratic party was awarded 3 seats through the means expected and the logic in the audit file is correct.	

3	Verify that the election process in resultFile.txt	resultFile.txt	The Democratic party should be allocated 2 seats by meeting quota and one by remainde	The Democratic party is awarded 3 seats through the means expected	
4					
5					

Project Name: Project 1 Voting System

Team #9

Test Stage: Unit __ System X

Test Date: 5/1/22

Test Case ID#: IRV_System_1

Name(s) of Testers: John Cullom

Test Description: Test to see if IRV system can successfully determine the winners for an election with a general case election scenario.

Indicate where are you storing the tests (what file) and the name of the method/functions being used.

Input_IRV.csv (run according to ReadMe instructions)

Automated: yes___ no X

Results: Pass X Fail

Preconditions for Test: Input file must be in the folder as the system's java files so that system can access it.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Place the Inout_IRV.csv file in testing/InputFiles from the testing folder. Navigate to the Project1 folder and type in "javac -d . src/*.java" And then type in	Input_IRV.csv			

	“java p1.Main testing/InputFiles”				
2	Verify that result.file has the correct election results	resultFile.txt	Rosen and Chou are declared winners of the election	Rosen and Chou are declared winners of the election	
3	Verify that the election process in audit.file	auditFile.txt	The election process can be followed and there are no noticeable errors. Rosen should be declared a winner after reaching droop during the first pass. Chou should be declared a winner by being promoted from loser list.	Rosen is chosen by droop after receiving three votes, Chou is promoted to the winners list after all candidates have either won or lost.	There should be no ballot reallocation in this test as the votes go 3,2,1,0 for Rosen, Chou, Royce, Klienberg.

Post condition(s) for Test: New audit and result files are generated and system successfully finishes running.

Project Name: Project 1 Voting System

Team #9

Test Stage: Unit __ System X

Test Date: 5/1/22

Test Case ID#: IRV_System_2

Name(s) of Testers: John Cullom, Anna Frenz

Test Description: Test to see if IRV system can recognize that a candidate reached droop and declare them a winner

Indicate where are you storing the tests (what file) and the name of the method/functions being used.

IRV_Input1.csv (run according to ReadMe instructions)

Automated: yes___ no X

Results: Pass X Fail

Preconditions for Test: Input file must be in the folder as the system's java files so that system can access it.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Place the Inout_IRV1.csv file in testing/InputFiles from the testing folder. Navigate to the Project1 folder and type in “javac -d . src/*.java” And then type in “java p1.Main testing/InputFiles”	Input_IRV1.csv			
2	Verify that result.file has the correct election results	resultFile.txt	Rosen and Klienberg are declared winners of the election	Rosen and Klienberg are declared winners of the election	
3	Verify that the election process in audit.file	auditFile.txt	The election process can be followed and there are no noticeable errors. Rosen should be declared a winner by reaching the droop quota and Klienberg should be declared a winner	Rosen was declared a winner during the first pass of the ballots for reaching droop. Klienberg reached droop from a reallocation of votes, making him the second winner. The election ended correctly.	

			by reaching droop after reallocation of votes from Klienberg, Royce		

Post condition(s) for Test: New audit and result files are generated and system successfully finishes running.

Project Name: Project 1 Voting System

Team #9

Test Stage: Unit __ System X **Test Date: 5/1/22**

Test Case ID#: IRV_System_3 **Name(s) of Testers: John Cullom**

Test Description: Test to see if different candidates win depending on election run if when there is a tie

Automated: yes___ no X **Indicate where are you storing the tests (what file) and the name of the method/functions being used.**

	Input_IRV2.csv (run according to ReadMe instructions)
Results: Pass X Fail	

Preconditions for Test: Input file must be in the folder as the system's java files so that system can access it.	

Step #	**Test Step Description**	**Test Data**	**Expected Result**	**Actual Result**	**Notes**
1	Place the Inout_IRV2.csv file in testing/InputFiles from the testing folder. Navigate to the Project1 folder and	Input_IRV2.csv			

	type in “javac -d . src/*.java” And then type in “java p1.Main testing/InputFiles”				
2	Verify that result.file has the correct election results	resultFile.txt	Rosen is declared winner of the election. Either Chou or Kleinberg is declared second winner based on the order of the vote processing	Rosen and Chou are winners.	
3	Verify that the election process in audit.file	auditFile.txt	The election process can be followed and there are no noticeable errors. Rosen should be declared a winner by reaching the droop quota and either Klienberg or Chou should be declared a winner by promoting them from loser list.	Rosen is declared a winner by reaching droop quota and Chou is declared a winner by promoting them from the loser list.	
4	Repeat steps 1				
5	Repeat step 2	resultFile.txt	Klienberg will be declared a winner instead of Chou	Klienberg is declared a winner instead of Chou	

Post condition(s) for Test: New audit and result files are generated and system successfully finishes running.

Project Name: Project 1 Voting System

Team #9

Test Stage: Unit __ System X	**Test Date: 5/1/22**
Test Case ID#: IRV_System_4	**Name(s) of Testers: John Cullom, Anna Frenz**
Test Description: Test to see if system will recognize that candidate reached droop quota through reallocation	
	Indicate where are you storing the tests (what file) and the name of the method/functions being used. **Input_IRV3.csv (run according to ReadMe instructions)**
Automated: yes___ no X	

Results: Pass X Fail

Preconditions for Test: Input file must be in the folder as the system's java files so that system can access it.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Place the Inout_IRV3.csv file in testing/InputFiles from the testing folder. Navigate to the Project1 folder and type in "javac -d . src/*.java" And then type in "java p1.Main testing/InputFiles"	Input_IRV3.csv			
2	Verify that result.file has the	resultFile.txt	Rosen and Chou are winners	Rosen and Chou are winners.	

	correct election results				
3	Verify that the election process in audit.file	auditFile.txt	The election process can be followed and there are no noticeable errors. Rosen is declared winner of the election by reaching droop quota through reallocation. Chou is declared the second winner of the election by appointing the last loser. Klienberg and Royce should tie in the first pass when determining the loser, but both will end up losing.	Rosen is declared a winner after getting vote 3 reallocated to him and Chou is appointed winner from the loser list.	
4					
5					

Post condition(s) for Test: New audit and result files are generated and system successfully finishes running

---

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:
PBI 7: Task 7: Run System Integration Tests. Test_7_1

Team Member(s) Responsible: Anna Frenz

Inputs: Ran an IRV election by following the instructions with InputIRV.csv and InputIRV1.csv files in InputFiles folder

Tests: Verify that audit file, result file, and outputted results were correct

Outputs: Audit file details process as expected including Rosen reaching droop and Chou being appointed a winner after votes are redistributed. The result file and result output have Rosen and Chou as winners as expected

Passed or Failed: Passed

Date: 4/27/22

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:
PBI 7: Task 7: Run System Integration Tests. Test_7_2

Team Member(s) Responsible: Anna Frenz

Inputs: Ran an OPL election by following the instructions with Input_OPL.csv and Input_OPL1.csv files in InputFiles folder

Tests: Verify that audit file, result file, and outputted results were correct

Outputs: Received Error when running election
“Exception in thread "main" java.lang.IndexOutOfBoundsException: Index 11 out of bounds for length 1”

Passed or Failed: Failed

Date: 4/27/22

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:
PBI 7: Task 7: Run System Integration Tests. Test_7_3

Team Member(s) Responsible: Anna Frenz

Inputs: Ran an OPL election by following the instructions with Input_OPL1.csv and Input_OPL2.csv files in InputFiles folder

Tests: Verify that audit file, result file, and outputted results were correct

Outputs: Received Error when running election
“Exception in thread "main" java.lang.IndexOutOfBoundsException: Index 11 out of bounds for length 1”

Passed or Failed: Failed

Date: 4/27/22

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:
PBI 8: Task 2: write code to remove ballots Test_8_1

Team Member(s) Responsible: Kyle Bekken

Inputs: Ran an IRV election by following the instructions with Input_IRV.csv, Input_IRV1.csv, and Input_IRV2.csv, Input_IRV3.csv files in InputFiles folder

Tests: Verify that all IRV ballots have at least half of candidates are voted for. Ifodd number of Candidates, round up.

Outputs: Rosen (D) and Chou (I) won the election via chart printed

Passed or Failed: Passed

Date: 4/27/22

Date: 4/27/22

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:
PBI 8: Task 3: write code to write ballots to invalid ballot file Test_8_2

Team Member(s) Responsible: Kyle Bekken

Inputs: Ran an IRV election by following the instructions with Input_IRV.csv, Input_IRV1.csv, and Input_IRV2.csv, Input_IRV3.csv files in InputFiles folder

Tests: If all “invalid” ballots are written to a file

Outputs: all “invalid” ballots were written to invalidBallots.txt in the misc folder

Passed or Failed: Passed

Date: 4/27/22

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:
PBI 7: Task 7: Run System Integration Tests. Test_7_4

Team Member(s) Responsible: Anna Frenz

Inputs: Ran an OPL election by following the instructions with Input_OPL1.csv, Input_OPL4.csv and Input_OPL2.csv files in InputFiles folder

Tests: Verify that audit file, result file, and outputted results were correct

Outputs: Results, audit file, and result file were all as expected.

Passed or Failed: Passed

Date: 4/28/22

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:

PBI 7: Task 7: Run System Integration Tests. Test_7_5

Team Member(s) Responsible: Anna Frenz

Inputs: Ran an IRV election by following the instructions with Input_IRV.csv, Input_IRV1.csv, Input_IRV2.csv, Input_IRV3.csv in Inputs folder
Tests: Verify that audit file, result file, and outputted results were correct

Outputs: Results, audit file, and result file were all as expected.

Passed or Failed: Passed

Date: 4/28/22

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:
PBI 7: Task 7: Run System Integration Tests. Test_7_6

Team Member(s) Responsible: Anna Frenz

Inputs: Ran a PO election by following the instructions with Input_PO.csv and Input_PO2.csv in Inputs folder

Tests: Verify that outputted results were correct

Outputs: Outputted result that Pike won with 5 votes was as expected.

Passed or Failed: Passed

Date: 4/28/22

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:
PBI 1: Task 4: Run System Integration Tests. Test_1_1

Team Member(s) Responsible: Anna Frenz

Inputs: Ran a PO election by following the instructions with Input_PO.csv in Inputs folder

Tests: Verify that audit file and outputted results were correct

Outputs: Outputted result that Pike won with 3 votes, audit file were as expected.

Passed or Failed: Passed

Date: 5/1/22

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:
PBI 1: Task 4: Run System Integration Tests. Test_1_2

Team Member(s) Responsible: Anna Frenz

Inputs: Ran a PO election by following the instructions with Input_PO1.csv in Inputs folder

Tests: Verify that audit file and outputted results were correct

Outputs: Outputted result that Smith won with 4 votes, audit file were as expected.

Passed or Failed: Passed

Date: 5/1/22

The PBI, the Task Description (from Sprint Log) with Unique Testing Number:
PBI 1: Task 4: Run System Integration Tests. Test_1_3

Team Member(s) Responsible: Anna Frenz

Inputs: Ran a PO election by following the instructions with Input_PO.csv and Input_PO2.csv in Inputs folder

Tests: Verify that audit file and outputted results were correct

Outputs: Outputted result that Pike won with 5 votes and audit file are as expected.

Passed or Failed: Passed

Date: 5/1/22