Project Name: Project 1: Voting System **Team 6**

Test Stage: Unit **Test Date: 11/13/2018**

Test Case ID#: IRV_Constructor_Test
Test Description: Test to make sure the default constructor for the InstantRunoffBallot class works correctly

Name(s) of Testers: John Caspers

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

Automated: yes

Results: Pass

Preconditions for Test: Instances of InstantRunoffBallot can successfully be created

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Create IRVballot object ballot	Test IR ballot	New ArrayList<>() == ballot	New ArrayList<>() == ballot	
3					
4					

Post condition(s) for Test:
Returns pass signifying that a InstantRunoffBallot was created correctly

Test Stage: Unit **Test Date: 11/13/18**

Test Case ID#: IRV_GetVotesList_ **Name(s) of Testers: David**

Test Description: Test to make sure GetVoteList() works and returns a vote list

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

Automated: yes

Results: Pass

Preconditions for Test: Instances of InstantRunoffBallot can successfully be created

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Create InstantRunoffBallot ballot				
3	assertEquals()	createVoteList(), ballot.getVoteList()	Both are equal	Both are equal	
4					

Post condition(s) for Test: The test passes and createVoteList() equals ballot.getVoteList()

Test Stage: Unit

Test Date: 11/13/18

Test Case ID#: IRV_testEquals_
Test Description: Test Equals() method of InstantRunoffBallot class

Name(s) of Testers: David

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

Automated: yes

Results: Pass

Preconditions for Test: Instances of InstantRunoffBallot can successfully be created

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Create InstantRunoffBallot ballot1				
3	Create InstantRunoffBallot ballot2				
4	Create InstantRunoffBallot ballot3				
5	assertEquals() assertEquals() assertNotEquals()	(ballot1, ballot1) (ballot1, ballot2) (ballot1, ballot3)	ballot1 == ballot1 ballot1 == ballot2 ballot1 != ballot3	ballot1 == ballot1 ballot1 == ballot2 ballot1 != ballot3	

Post condition(s) for Test: ballot1 is equal to ballot2. Ballot1 is not equal to ballot3

Test Stage: Unit **Test Date: 11/13/18**

Test Case ID#: OPL_Constructor_ **Name(s) of Testers: David**

Test Description: Test to make sure the OpenPartyListBallot class objects can successfully be created

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

Automated: yes

Results: Pass

Preconditions for Test: Instances of OpenParty can successfully be created

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL ballot object		Ballot.getvoteList() == newArrayList<>()	Ballot.getvoteList() == newArrayList<>()	
2					

Post condition(s) for Test: assertEquals(createVoteList(candidateNum, party), ballot.getVoteList()) returns true

Test Stage: Unit **Test Date: 11/13/18**

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

Test Description: Test to make sure the OpenPartyListBallot class objects can successfully be created and GetVoteList()

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

Automated: yes

Results: Pass

Preconditions for Test: Instances of OpenParty can successfully be created and GetVoteList() works

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL ballot object		Ballot.getvoteList() == newArrayList<>()	Ballot.getvoteList() == newArrayList<>()	
2					

Post condition(s) for Test: assertEquals(createVoteList(candidateNum, party), ballot.getVoteList()) returns true

Test Stage: Unit **Test Date: 11/13/18**

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

Test Description: Test to make sure the OpenPartyListBallot class objects can successfully be created and Equals() works

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

Automated: yes

Results: Pass

Preconditions for Test: Instances of OpenParty can successfully be created and Equals() works

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL ballot object		Ballot.getvoteList() == newArrayList<>()	Ballot.getvoteList() == newArrayList<>()	
2	Create InstantRunoffBallot ballot1				
3	Create InstantRunoffBallot ballot2				
4	Create InstantRunoffBallot ballot3				
5	assertEquals() assertEquals() assertNotEquals()	(ballot1, ballot1) (ballot1, ballot2) (ballot1, ballot3)	ballot1 == ballot1 ballot1 == ballot2 ballot1 != ballot3	ballot1 == ballot1 ballot1 == ballot2 ballot1 != ballot3	

Post condition(s) for Test: assertEquals(createVoteList(candidateNum, party), ballot.getVoteList()) returns true

Test Stage: System

Test Date: 11/13/18

Test Case ID#: OPL_runElection _

Name(s) of Testers: John

Test Description: Test to make sure an OPL runElection() works for a given OPL csv file

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

Automated: no

Results: Pass

Preconditions for Test: Input testOPLInput.cvs file path

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run Drivers.java				
3	Give testOPLInput.csv file path as input				
4	Give audit file output path				
5	Check output file		Audit file created and winners outputted to screen	Audit file created and winners outputted to screen	

Post condition(s) for Test: Pike, Foster, Borg are win and audit file was successfully created for given output path

Test Stage: System

Test Date: 11/13/18

Test Case ID#: IRV_runElection _

Name(s) of Testers: John

Test Description: Test to make sure an IRV runElection() works for a given IRV csv file

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

Automated: no

Results: Pass

Preconditions for Test: Input testOPLInput.cvs file path

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1					
2	Run Drivers.java				
3	Give testIRVInput.csv file path as input				
4	Give audit file output path				
5	Check output file		Audit file created and winners outputted to screen	Audit file created and winners outputted to screen	

Post condition(s) for Test: Chou wins and audit file was successfully created for given output path

Test Stage: Unit _x_ System __	**Test Date: 11/13/2018**
Test Case ID#: IRV_Vote_1	**Name(s) of Testers: Daniel Rockcastle**
Test Description: Tests default constructor for IRV.	
Automated: yes_x_ no ___	**Indicate where are you storing the tests (what file) and the name of the method/functions being used.** Stored in the class InstantRunoffVoteTest. The method is testDefaultConstructor.
Results: Pass ___x__ Fail_____	
Preconditions for Test: Instance of InstantRunoffVote can be created.	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Run the test.	N/A	Created vote has a default candidate and rank.	Created vote has a default candidate and rank.	
2					
3					
4					

Post condition(s) for Test:

Unit test passes.

Project Name: Project 1: Voting System	**Team#**
Test Stage: Unit _x_ System __	**Test Date: 11/13/2018**
Test Case ID#: IRV_Vote_2	**Name(s) of Testers: Daniel Rockcastle**
Test Description: Tests constructor for IRV.	

Automated: yes_x_ no ___	**Indicate where are you storing the tests (what file) and the name of the method/functions being used.** Stored in the class InstantRunoffVoteTest. The method is testConstructor.
Results: Pass ___x__ Fail_____	
Preconditions for Test: Instance of InstantRunoffVote can be created.	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Run the test.	N/A	Created vote has passed in candidate and passed in rank.	Created vote has passed in candidate and passed in rank.	
2					
3					
4					

Post condition(s) for Test:

Unit test passes.

Project Name: Project 1: Voting System	**Team#**
Test Stage: Unit _x_ System __	**Test Date: 11/13/2018**
Test Case ID#: IRV_Vote_3	**Name(s) of Testers: Daniel Rockcastle**
Test Description: Tests getCandidate for IRV.	
Automated: yes_x_ no ___	**Indicate where are you storing the tests (what file) and the name of the method/functions being used.** Stored in the class InstantRunoffVoteTest. The method is testGetCandidate.

Results: Pass ___x__ Fail_____	
Preconditions for Test: Instance of InstantRunoffVote can be created.	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Run the test.	N/A	Created vote can have candidate name and party pulled from it.	Created vote can have candidate name and party pulled from it.	
2					
3					
4					

Post condition(s) for Test:

Unit test passes.

Project Name: Project 1: Voting System	**Team#**
Test Stage: Unit _x_ System __	**Test Date: 11/13/2018**
Test Case ID#: IRV_Vote_4	**Name(s) of Testers: Daniel Rockcastle**
Test Description: Tests getRank for IRV.	
Automated: yes_x_ no ___	**Indicate where are you storing the tests (what file) and the name of the method/functions being used.** Stored in the class InstantRunoffVoteTest. The method is testGetRank.
Results: Pass ___x__ Fail_____	
Preconditions for Test: Instance of InstantRunoffVote can be created.	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Run the test.	N/A	Created vote can have rank pulled from it.	Created vote can have rank pulled from it.	
2					
3					
4					

Post condition(s) for Test:

Unit test passes.

Project Name: Project 1: Voting System	**Team#**
Test Stage: Unit _x_ System __	**Test Date: 11/13/2018**
Test Case ID#: IRV_Vote_5	**Name(s) of Testers: Daniel Rockcastle**
Test Description: Tests Equals for IRV.	
Automated: yes_x_ no ___	**Indicate where are you storing the tests (what file) and the name of the method/functions being used.** Stored in the class InstantRunoffVoteTest. The method is testEquals.
Results: Pass ___x__ Fail_____	
Preconditions for Test: Instance of InstantRunoffVote can be created.	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Run the test.	N/A	Two equally created votes are returned as equal.	Two equally created votes are returned as equal.	
2					

3					
4					

Post condition(s) for Test:

Unit test passes.

Project Name: Project 1: Voting System	**Team#**
Test Stage: Unit _x_ System __	**Test Date: 11/13/2018**
Test Case ID#: OPL_Vote_1	**Name(s) of Testers: Daniel Rockcastle**
Test Description: Tests default constructor for OPL.	
Automated: yes_x_ no ___	**Indicate where are you storing the tests (what file) and the name of the method/functions being used.** Stored in the class OpenPartyListVoteTest. The method is testDefaultConstructor.
Results: Pass ___x__ Fail_____	
Preconditions for Test: Instance of OpenPartyListVote can be created.	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Run the test.	N/A	Created vote has a default candidate.	Created vote has a default candidate.	
2					
3					
4					

Post condition(s) for Test:

Unit test passes.

Project Name: Project 1: Voting System	**Team#**
Test Stage: Unit _x_ System __	**Test Date: 11/13/2018**
Test Case ID#: OPL_Vote_2	**Name(s) of Testers: Daniel Rockcastle**
Test Description: Tests constructor for OPL.	
Automated: yes_x_ no ___	**Indicate where are you storing the tests (what file) and the name of the method/functions being used.** Stored in the class OpenPartyListVoteTest. The method is testConstructor.
Results: Pass ___x__ Fail_____	
Preconditions for Test: Instance of OpenPartyListVote can be created.	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Run the test.	N/A	Created vote has a candidate. Party and name check out.	Created vote has a candidate. Party and name check out.	
2					
3					
4					

Post condition(s) for Test:

Unit test passes.

Project Name: Project 1: Voting System	**Team#**
Test Stage: Unit _x_ System __	**Test Date: 11/13/2018**
Test Case ID#: OPL_Vote_3	**Name(s) of Testers: Daniel Rockcastle**
Test Description: Tests getCandidate for OPL.	

Automated: yes_x_ no ___	**Indicate where are you storing the tests (what file) and the name of the method/functions being used.** Stored in the class OpenPartyListVoteTest. The method is testGetCandidate.
Results: Pass ___x__ Fail_____	
Preconditions for Test: Instance of OpenPartyListVote can be created.	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Run the test.	N/A	Created vote has a candidate. Party and name check out.	Created vote has a candidate. Party and name check out.	
2					
3					
4					

Post condition(s) for Test:

Unit test passes.

Project Name: Project 1: Voting System	**Team#**
Test Stage: Unit _x_ System __	**Test Date: 11/13/2018**
Test Case ID#: OPL_Vote_4	**Name(s) of Testers: Daniel Rockcastle**
Test Description: Tests equals for OPL.	
Automated: yes_x_ no ___	**Indicate where are you storing the tests (what file) and the name of the method/functions being used.** Stored in the class OpenPartyListVoteTest. The method is testEquals.
Results: Pass ___x__ Fail_____	

Preconditions for Test: Instance of OpenPartyListVote can be created.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Run the test.	N/A	Two equally created OPL votes are checked to be equal.	Two equally created OPL votes are checked to be equal.	
2					
3					
4					

Post condition(s) for Test:

Unit test passes.

Test Stage: Unit _X_ System __ **Test Date: 11/11/2018**

Test Case ID#: CandidateTest_1
Test Description: Tests the default constructor in Candidate.java

Name(s) of Testers: Daniel Rockcastle

Automated: yes_x__ no ___

Indicate where are you storing the tests (what file) and the name of the method/functions being used.
testDefaultConstructor() function in CandidateTest.java in testing folder.

Results: Pass _x_ Fail_____

Preconditions for Test:
The code compiles.

Step	Test Step	Test	Expected	Actual	

#	Description	Data	Result	Result	Notes
1	Checks the candidate name, candidate party name and number of votes initially (=0)	N/A	True	True	
2					
3					
4					

Post condition(s) for Test:

The test passes.

Test Stage: Unit _X_ System __

Test Date: 11/11/2018

Test Case ID#: CandidateTest_3

Name(s) of Testers: Daniel Rockcastle

Test Description: Tests the getName() function in Candidate.java

Indicate where are you storing the tests (what file) and the name of the method/functions being used.
testGetName() function in CandidateTest.java in testing folder.

Automated: yes_x__ no ___

Results: Pass _x_ Fail_____

Preconditions for Test:
The code compiles.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Checks the candidate name	n/a	True	True	
2					
3					

4					

Post condition(s) for Test:

The test passes.

Test Stage: Unit _X_ System __ **Test Date: 11/11/2018**

Test Case ID#: CandidateTest_4 **Name(s) of Testers: Daniel Rockcastle**

Test Description: Tests the getNumberOfVotes function in Candidate.java

Automated: yes_x__ no ___ **Indicate where are you storing the tests (what file) and the name of the method/functions being used. testGetNumberOfVotes() function in CandidateTest.java in testing folder.**

Results: Pass _x_ Fail_____

Preconditions for Test:
The code compiles.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Checks the number of votes	n/a	`True`	`True`	
2					
3					
4					

Post condition(s) for Test:

The test passes.

Test Stage: Unit _X_ System __ **Test Date: 11/11/2018**

Test Case ID#: CandidateTest_5 **Name(s) of Testers: Daniel Rockcastle**

Test Description: Tests the addVote() function

Indicate where are you storing the tests (what file) and the name of the method/functions being used.
testAddVote() function in CandidateTest.java in testing folder.

Automated: yes_x__ no ___

Results: Pass _x_ Fail_____

Preconditions for Test:
The code compiles.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Checks if the addVote increments the number of votes by 1	n/a	`True`	`True`	
2					
3					
4					

Post condition(s) for Test:

The test passes.

Test Stage: Unit _X_ System __ **Test Date: 11/11/2018**

Test Case ID#: CandidateTest_2 **Name(s) of Testers: Daniel Rockcastle**

Test Description: Tests the parametrized constructor in Candidate.java

Indicate where are you storing the tests (what file) and the name of the method/functions being used.
testConstructor() function in CandidateTest.java in testing folder.

Automated: yes_x__ no ___

Results: Pass _x_ Fail_____

Preconditions for Test:
The code compiles.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Checks the candidate name, candidate party name and number of votes		`True`	`True`	
2					
3					
4					

Post condition(s) for Test:

The test passes.

Test Stage: Unit _✓_ System __

Test Date: 11/13/2018

Test Case ID#: ReadInputTest_1

Name(s) of Testers: Michael Birk

Test Description: Tests the parseInputLinesOPL() function

Indicate where are you storing the tests (what file) and the name of the method/functions being used.
testParseInputLinesOPL() in ReadInputTest.java in testing folder.

Automated: yes_✓_ no ___

Results: Pass: Yes Fail_____

Preconditions for Test: The code compiles.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Run the test	N/A	Two equally created OpenPartyListElection objects are created.	Two equally created OpenPartyListElection objects are created.	
2					
3					
4					

Post condition(s) for Test:
The test passes

Test Stage: Unit _X_ System __

Test Date: 11/11/2018

Test Case ID#: CandidateTest_7

Name(s) of Testers: Daniel Rockcastle

Test Description: Tests the getParty() function

Indicate where are you storing the tests (what file) and the name of the method/functions being used.
testGetParty() function in CandidateTest.java in testing folder.

Automated: yes_x__ no ___

Results: Pass _x_ Fail_____

Preconditions for Test:
The code compiles.

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

1	Checks if getParty function returns the correct party name	N/A	True	True	
2					
3					
4					

Post condition(s) for Test:

The test passes.

Test Stage: Unit _X_ System __

Test Date: 11/11/2018

Test Case ID#: CandidateTest_6

Name(s) of Testers: Daniel Rockcastle

Test Description: Tests the equals() function

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

Automated: yes_x__ no ___

testEquals() function in CandidateTest.java in testing folder.

Results: Pass _x_ Fail_____

Preconditions for Test:
The code compiles.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Checks if the equals function works properly	N/A	True	True	
2					
3					
4					

Post condition(s) for Test:

The test passes.

Test Stage: Unit _X_ System __ **Test Date: 11/13/2018**

Test Case ID#: ReadInputTest_2 **Name(s) of Testers: Michael Birk**

Test Description: Tests the parseInputLinesIRV() function.

Automated: yes_x__ no ___

Indicate where are you storing the tests (what file) and the name of the method/functions being used.
testParseInputLinesIRV() function in ReadInputTest.java testing folder.

Results: Pass _x_ Fail_____

Preconditions for Test:
The code compiles.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Run the test.	N/A	Two equally created InstantRunOffElection objects are created	Two equally created InstantRunOffElection objects are created	
2					
3					
4					

Post condition(s) for Test:

The test passes.