

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <u> X </u> System <u> </u>		Test Date: 03/14/2021
Test Case ID#: test_AuditLog_1		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates a test Audit log that prints initial ballot information, the processing of the ballot, and the results.		
Automated: yes <u> X </u> no <u> </u>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. FileCreation();
Results: Pass <u> X </u> Fail <u> </u>		<pre> [-----] 1 test from fixture_AuditLog [RUN] fixture_AuditLog.FileCreation [OK] fixture_AuditLog.FileCreation (0 ms) [-----] 1 test from fixture_AuditLog (0 ms total) </pre>
Preconditions for Test: The audit log is requested for the program		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Print out ballot information	Log()	Ballot information can be seen		
2	Print out ballot processing	Log()	Ballot processing can be viewed		
3	Print out election results	Log()	Election results are printed		
4					

Post condition(s) for Test:

The audit file is available to be read by the auditor

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <input checked="" type="checkbox"/> System <input type="checkbox"/>		Test Date: 03/14/2021
Test Case ID#: test_Ballot_1		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates a unique id for each ballot		
Automated: yes <input checked="" type="checkbox"/> no <input type="checkbox"/>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. BallotUniqueID
Results: Pass <input checked="" type="checkbox"/> Fail <input type="checkbox"/>		<pre>[-----] 2 tests from fixture_Ballot [RUN] fixture_Ballot.BallotUniqueID [OK] fixture_Ballot.BallotUniqueID (0 ms) [RUN] fixture_Ballot.BallotUniqueIDRepeat [OK] fixture_Ballot.BallotUniqueIDRepeat (78 ms) [-----] 2 tests from fixture_Ballot (78 ms total)</pre>
Preconditions for Test: Test ballot is created		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	create test ballot	TestBallot()	Ballot is created		
2	create a unique id for test balot	get_id()	Ballot receives an id		
3					
4					

Post condition(s) for Test:

The ballot can now be referenced by other parts of the program given ID

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <input checked="" type="checkbox"/> System <input type="checkbox"/>		Test Date: 03/14/2021
Test Case ID#: test_Ballot_2		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates database for all ballots after they have an id and generates a lot of ballots to test and make sure each id is unique		
Automated: yes <input checked="" type="checkbox"/> no <input type="checkbox"/>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. BallotUniqueIDRepeat
Results: Pass <input checked="" type="checkbox"/> Fail <input type="checkbox"/>		<pre> [-----] 2 tests from fixture_Ballot [RUN] fixture_Ballot.BallotUniqueID [OK] fixture_Ballot.BallotUniqueID (0 ms) [RUN] fixture_Ballot.BallotUniqueIDRepeat [OK] fixture_Ballot.BallotUniqueIDRepeat (78 ms) [-----] 2 tests from fixture_Ballot (78 ms total) </pre>

Preconditions for Test: Each ballot has an assigned id	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	store ballots in a database	std::set<uint64_t> ids	Ballot id's are now stored in a database		
2	generate many ballots	TestBallot()	Many ballots are generated		
3	check to make sure generated and stored ballots are unique	get_id()	all ballots have unique id's		
4					

Post condition(s) for Test:

Ballots can now be accessed by other parts of the code via the created database

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <u> X </u> System <u> </u>		Test Date: 03/14/2021
Test Case ID#: test_IRBallot_1		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Tests the basic functionality of the IRBallotCSVParse function		
Automated: yes <u> X </u> no <u> </u>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. IRBallotCSVParse()
Results: Pass <u> X </u> Fail <u> </u>		[-----] 3 tests from fixture_IRBallot [RUN] fixture_IRBallot.IRBallotCSVParse

	<pre> [OK] fixture_IRBallot.IRBallotCSVParse (0 ms) [RUN] fixture_IRBallot.IRBallotIncrement [OK] fixture_IRBallot.IRBallotIncrement (0 ms) [RUN] fixture_IRBallot.IRBallotLog [OK] fixture_IRBallot.IRBallotLog (0 ms) [-----] 3 tests from fixture_IRBallot (0 ms total) </pre>
Preconditions for Test: Ballots have been created	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	create a new ballot with election data and store it	IRBallot(1,2,3,4)	ballot information is stored correctly		
2	extract the ballot information	get_choice()	ballot information has been read correctly and prints correctly		
3					
4					

Post condition(s) for Test:

Ballot information can now be read and used by other parts of the code.

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <u> X </u> System <u> </u>		Test Date: 03/14/2021

Test Case ID#: test_IRBallot_2	Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Tests the basic functionality of the IRBallotIncrement function	
Automated: yes_X__ no ____	Indicate where are you storing the tests (what file) and the name of the method/functions being used. IRBallotIncrement()
Results: Pass __X__ Fail _____	<pre> [-----] 3 tests from fixture_IRBallot [RUN] fixture_IRBallot.IRBallotCSVParse [OK] fixture_IRBallot.IRBallotCSVParse (0 ms) [RUN] fixture_IRBallot.IRBallotIncrement [OK] fixture_IRBallot.IRBallotIncrement (0 ms) [RUN] fixture_IRBallot.IRBallotLog [OK] fixture_IRBallot.IRBallotLog (0 ms) [-----] 3 tests from fixture_IRBallot (0 ms total) </pre>
Preconditions for Test: Ballots have been created	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	create a new ballot with election data and store it	IRBallot(1,2,3,4)	ballot information is stored correctly		
2	extract the ballot information and increment the choice based on the ballot information	increment_choice();	ballot information has been read correctly and prints correctly		
3					
4					

Post condition(s) for Test:

Ballots can now be accessed by other parts of the code via the created database

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <input checked="" type="checkbox"/> System <input type="checkbox"/>		Test Date: 03/14/2021
Test Case ID#: test_IRBallot_3		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Tests the basic functionality of the IRBallotLog function		
Automated: yes <input checked="" type="checkbox"/> no <input type="checkbox"/>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. IRBallotLog()
Results: Pass <input checked="" type="checkbox"/> Fail <input type="checkbox"/>		<pre>[-----] 3 tests from fixture_IRBallot [RUN] fixture_IRBallot.IRBallotCSVParse [OK] fixture_IRBallot.IRBallotCSVParse (0 ms) [RUN] fixture_IRBallot.IRBallotIncrement [OK] fixture_IRBallot.IRBallotIncrement (0 ms) [RUN] fixture_IRBallot.IRBallotLog [OK] fixture_IRBallot.IRBallotLog (0 ms) [-----] 3 tests from fixture_IRBallot (0 ms total)</pre>
Preconditions for Test: Ballots have been created		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	create a new ballot with election data and store it	IRBallot(1,2,3,4)	ballot information is stored correctly		
2	stores ballot information	get_id()	ballot information has been read correctly and prints correctly		
3	takes stored ballot information and makes it accessible to log to the audit file	log()the	ballot information is printed to audit log		
4					

Post condition(s) for Test:

Audit log prints ballot information

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <input checked="" type="checkbox"/> System <input type="checkbox"/>		Test Date: 03/14/2021
Test Case ID#: test_IRelection_1		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Tests if the IRelection runs properly		
Automated: yes <input checked="" type="checkbox"/> no <input type="checkbox"/>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. IRelection()
Results: Pass <input checked="" type="checkbox"/> Fail <input type="checkbox"/>		<pre>[-----] 2 tests from fixture_IRelection [RUN fixture_IRelection.IRelectionSRSEExample The winner of the election is Rosen of party D with 50% of the votes Kleinberg had 0% of the votes. Chou had 33% of the votes. Royce had 16% of the votes.</pre>

	[OK] fixture_IRelection.IRelectionSRSEExample (1ms)
Preconditions for Test: IR Election has been selected	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	start new IRelection	IRelection()	new IR Election has been started		
2	run the IR Election	run();	IR Election runs without any errors		
3					
4					

Post condition(s) for Test:

Election results can be viewed from IR Election

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <u> X </u> System <u> </u>		Test Date: 03/14/2021
Test Case ID#: test_IRelection_2		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Tests if the IRelection runs properly with random values/candidates		
Automated: yes <u> X </u> no <u> </u>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. IRelection()
Results: Pass <u> X </u> Fail <u> </u>		[RUN]

	<pre>fixture_IRElection.IRElectionRandom The winner of the election is Candidate2Test of party 2 with 50% of the votes Candidate1Test had 43% of the votes. Candidate3Test had 44% of the votes. Candidate4Test had 40% of the votes. Candidate5Test had 43% of the votes. Candidate6Test had 42% of the votes. Candidate7Test had 45% of the votes. Candidate8Test had 44% of the votes. Candidate9Test had 44% of the votes. Candidate10Test1 had 4% of the votes. Candidate11Test1 had 40% of the votes. Candidate12Test1 had 39% of the votes. Candidate13Test1 had 45% of the votes. Candidate14Test1 had 44% of the votes. Candidate15Test1 had 39% of the votes. Candidate16Test1 had 45% of the votes. Candidate17Test1 had 48% of the votes. Candidate18Test1 had 46% of the votes. Candidate19Test1 had 41% of the votes. Candidate20Test2 had 49% of the votes. Candidate21Test2 had 45% of the votes. Candidate22Test2 had 42% of the votes. Candidate23Test2 had 49% of the votes. Candidate24Test2 had 40% of the votes. Candidate25Test2 had 47% of the votes. [OK] fixture_IRElection.IRElectionRandom (8253 ms)</pre>
Preconditions for Test: IR Election has been selected	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	start new IRElection	IRElection()	new IR Election has been started		
2	run the IR Election	run();	IR Election runs without any errors		

3					
4					

Post condition(s) for Test:

Election results can be viewed from IR Election

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <input checked="" type="checkbox"/> System <input type="checkbox"/>		Test Date: 03/14/2021
Test Case ID#: test_MediaReport_1		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Tests if information from the election and audit log can be printed in a media report		
Automated: yes <input checked="" type="checkbox"/> no <input type="checkbox"/>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. MediaReport()
Results: Pass <input checked="" type="checkbox"/> Fail <input type="checkbox"/>		<pre> [-----] 1 test from fixture_MediaReport [RUN] fixture_MediaReport.MediaReportFileCreation [OK] fixture_MediaReport.MediaReportFileCreation (1 ms) [-----] 1 test from fixture_MediaReport (1 ms total) </pre>
Preconditions for Test: The election has finished		

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

1	create media report	MediaReport()	A media		
2	write to the media report	write()	election results have been written to the media report and are correct		
3					
4					

Post condition(s) for Test:

Election results have been shared with the media

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <input checked="" type="checkbox"/> System <input type="checkbox"/>		Test Date: 03/14/2021
Test Case ID#: test_OPLBallot_1		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL ballot that is a CSV and parses a single choice from the ballot		
Automated: yes <input checked="" type="checkbox"/> no <input type="checkbox"/>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLBallot()
Results: Pass <input checked="" type="checkbox"/> Fail <input type="checkbox"/>		[RUN] fixture_OPLBallot.OPLBallotCSVParse_1choice [OK] fixture_OPLBallot.OPLBallotCSVParse_1choice (0 ms)
Preconditions for Test: OPL election is selected		

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

1	Create OPL Ballot	OPLBallot()	OPL Ballot has been created		
2	read information from ballot	get_choice()	ballot information has been read		
3					
4					

Post condition(s) for Test:

Ballot information can be viewed and used by other parts of the code

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <u> X </u> System <u> </u>		Test Date: 03/14/2021
Test Case ID#: test_OPLBallot_2		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL ballot that is a CSV and parses the file with no choices from the ballot		<pre>[RUN] fixture_OPLBallot.OPLBallotCSVParse_0choice [OK] fixture_OPLBallot.OPLBallotCSVParse_0choice (0 ms)</pre>
Automated: yes <u> X </u> no <u> </u>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLBallot()
Results: Pass <u> X </u> Fail <u> </u>		
Preconditions for Test: OPL election is selected		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL Ballot	OPLBallot()	OPL Ballot has been created		
2	read information from ballot	get_choice()	ballot information has been read		
3					

4					

Post condition(s) for Test:

Ballot information can be viewed and used by other parts of the code

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <u> X </u> System <u> </u>		Test Date: 03/14/2021
Test Case ID#: test_OPLBallot_3		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL ballot and logs a single choice from the ballot		
Automated: yes <u> X </u> no <u> </u>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLBallot()
Results: Pass <u> X </u> Fail <u> </u>		[RUN] fixture_OPLBallot.OPLBallotLogOnceChoice [OK] fixture_OPLBallot.OPLBallotLogOnceChoice (0 ms)
Preconditions for Test: OPL election is selected		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL Ballot	OPLBallot()	OPL Ballot has been created		
2	retrieve ballot with information	get_id()	ballot has been selected and information has been read		
3	log ballot infromation	log()	ballot information can be viewed in the audit log		
4					

Post condition(s) for Test:

Audit log results are printed

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <input checked="" type="checkbox"/> System <input type="checkbox"/>		Test Date: 03/14/2021
Test Case ID#: test_OPLBallot_4		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL ballot and logs a no choices from the ballot		
Automated: yes <input checked="" type="checkbox"/> no <input type="checkbox"/>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLBallot()
Results: Pass <input checked="" type="checkbox"/> Fail <input type="checkbox"/>		[RUN] fixture_OPLParty.OPLParty_1party_1candidate [OK] fixture_OPLParty.OPLParty_1party_1candidate (0 ms)
Preconditions for Test: OPL election is selected		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL Ballot	OPLBallot()	OPL Ballot has been created		
2	retrieve ballot with information	get_id()	ballot has been selected and information has been read		
3	log ballot information	log()	ballot information can be viewed in the audit log		
4					

Post condition(s) for Test:

Audit log results are printed

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <input checked="" type="checkbox"/> System <input type="checkbox"/>		Test Date: 03/14/2021
Test Case ID#: test_OPLCandidate_1		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an empty OPLCandidate		
Automated: yes <input checked="" type="checkbox"/> no <input type="checkbox"/>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLCandidate()
Results: Pass <input checked="" type="checkbox"/> Fail <input type="checkbox"/>		[RUN] fixture_OPLCandidate.OPLCandidate_empty_log [OK] fixture_OPLCandidate.OPLCandidate_empty_log (0 ms)
Preconditions for Test: OPL election is selected		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL Candidate	OPLCandidate()	OPL Candidate has been created		
2	log candidate information	log()	Candidate information can be viewed from the audit log - expected to be of the form: "Candidate <X>: ballots = []"		

Post condition(s) for Test:

Audit log results are printed

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <u> X </u> System <u> </u>		Test Date: 03/14/2021
Test Case ID#: test_OPLCandidate_2		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL candidate with no votes		
Automated: yes <u> X </u> no <u> </u>		Indicate where you are storing the tests (what file) and the name of the method/functions being used. OPLCandidate
Results: Pass <u> X </u> Fail <u> </u>		[RUN] fixture_OPLCandidate.OPLCandidate_empty_gettally [OK] fixture_OPLCandidate.OPLCandidate_empty_gettally (0 ms)
Preconditions for Test: OPL election is selected		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL candidate	OPLCandidate()	OPL candidate has been created		
2	retrieve the number of votes given to a candidate	get_tally()	the number of votes that have been read for a particular candidate - 0	0	
3					
4					

Post condition(s) for Test:

Candidate wins election

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <u>X</u> System <u> </u>		Test Date: 03/14/2021
Test Case ID#: test_OPLCandidate_3		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL candidate who has votes		
Automated: yes <u>X</u> no <u> </u>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLCandidate
Results: Pass <u>X</u> Fail <u> </u>		[RUN] fixture_OPLCandidate.OPLCandidate_gettally [OK] fixture_OPLCandidate.OPLCandidate_gettally (0 ms)
Preconditions for Test: OPL election is selected		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL candidate	OPLCandidate()	OPL candidate has been created		
2	retrieve the number of votes given to a candidate	log()	the number of votes that have been read for a particular candidate		
3					
4					

Post condition(s) for Test:

Candidate wins election

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <u> X </u> System <u> </u>		Test Date: 03/14/2021
Test Case ID#: test_OPL_Party_1		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL party with candidates in the party		
Automated: yes <u> X </u> no <u> </u>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLParty()
Results: Pass <u> X </u> Fail <u> </u>		[RUN] fixture_OPLParty.OPLParty_getname [OK] fixture_OPLParty.OPLParty_getname (0 ms)
Preconditions for Test: OPL election is selected		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL party	OPLParty()	OPL party has been created		
2	retrieve the names of the candidates in the party	get_name()	the name of the candidates in the party has been received		
3					
4					

Post condition(s) for Test:

Candidates party wins the election

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <u> X </u> System <u> </u>		Test Date: 03/14/2021
Test Case ID#: test_OPL_Party_2		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL party with candidates in the party and votes for each candidate and party		
Automated: yes <u> X </u> no <u> </u>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLParty()
Results: Pass <u> X </u> Fail <u> </u>		[RUN] fixture_OPLParty.OPLParty_gettally [OK] fixture_OPLParty.OPLParty_gettally (0 ms)
Preconditions for Test: OPL election is selected		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL party	OPLParty()	OPL party has been created		
2	retrieve votes of each candidate in a party	get_tally()	votes for each candidate have been read		
3					
4					

Post condition(s) for Test:

Candidates party wins the election

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <u> X </u> System <u> </u>		Test Date: 03/14/2021
Test Case ID#: test_OPL_Party_3		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL party with candidates in the party and votes for each candidate and party		
Automated: yes <u> X </u> no <u> </u>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLParty()
Results: Pass <u> X </u> Fail <u> </u>		[RUN] fixture_OPLParty.OPLParty_log [OK] fixture_OPLParty.OPLParty_log (0 ms)
Preconditions for Test: OPL election is selected		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL party	OPLParty()	OPL party has been created		
2	Retrieve record of candidates and ballots associated with party	log()	Votes for specific candidates in a party are received and displayed	"OPL Party Democrat: []"	
3					
4					

Post condition(s) for Test:

Candidates party wins the election

Project Name: Project 1: Voting System	Team# 20
---	-----------------

Test Stage: Unit _X_ System __	Test Date: 03/14/2021
Test Case ID#: test_OPL_Party_4	Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL party with candidates in the party	
Automated: yes_X_ no __	Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLParty()
Results: Pass __X__ Fail _____	[RUN] fixture_OPLParty.OPLParty_1party_1candidate [OK] fixture_OPLParty.OPLParty_1party_1candidate (0 ms)
Preconditions for Test: OPL election is selected	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL party	OPLParty()	OPL party has been created		
2	retrieve the votes of the candidates in the party	get_tally()	the name of the candidates in the party has been received		
3					
4					

Post condition(s) for Test:

Candidates party wins the election

Project Name: Project 1: Voting System	Team# 20
Test Stage: Unit _X_ System __	Test Date: 03/14/2021

Test Case ID#: test_OPL_Party_5	Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL party with 1 candidate, 1 party, and 3 ballots	
Automated: yes_X__ no ____	Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLParty()
Results: Pass_X__ Fail____	[RUN] fixture_OPLParty.OPLParty_1party_1candidate_3ballots [OK] fixture_OPLParty.OPLParty_1party_1candidate_3ballots (0 ms)
Preconditions for Test: OPL election is selected	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL party	OPLParty()	OPL party has been created		
2	create new candidate	OPLCandidate()	name of the candidate is read		
3	add the candidate to a party	add_candidate()	candidate has been added to a party		
4	create ballots with voter information	OPLBallot()	ballots have been created		
5	add ballots to a specific candidate based on	add_ballot()	ballots have been correctly assigned to the candidate		

Post condition(s) for Test:

Candidates party wins the election

Project Name: Project 1: Voting System

Team# 20

Test Stage: Unit <u> X </u> System <u> </u>	Test Date: 03/14/2021
Test Case ID#: test_OPL_Election_1	Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL party with 1 candidate, 1 party, and 3 ballots	
Automated: yes <u> X </u> no <u> </u>	Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLParty()
Results: Pass <u> X </u> Fail <u> </u>	[RUN] fixture_OPLParty.OPLParty_1party_1candidate_3ballots [OK] fixture_OPLParty.OPLParty_1party_1candidate_3ballots (0 ms)
Preconditions for Test: OPL election is selected	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL party	OPLParty()	OPL party has been created		
2	create new candidate	OPLCandidate()	name of the candidate is read		
3	add the candidate to a party	add_candidate()	candidate has been added to a party		
4	create ballots with voter information	OPLBallot()	information		
5	add ballots to a specific cand				

Post condition(s) for Test:

Candidates party wins the election

Test Stage: Unit <input checked="" type="checkbox"/> System <input type="checkbox"/>	Test Date: 03/14/2021
Test Case ID#: test_TieBreaker_1	Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Checks to see if the tie breaker function is functional in resolving ties	
Automated: yes <input checked="" type="checkbox"/> no <input type="checkbox"/>	Indicate where are you storing the tests (what file) and the name of the method/functions being used. TieBreaker()
Results: Pass <input checked="" type="checkbox"/> Fail <input type="checkbox"/>	[RUN] fixture_OPLParty.OPLParty_1party_1candidate_3ballots [OK] fixture_OPLParty.OPLParty_1party_1candidate_3ballots (0 ms)
Preconditions for Test: A tie in the election occurs	

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	creates a loop to go through coin flips and check to see if the function can resolve the tie properly	resolve_tie()	function resolves tie and assigns winner		
2					
3					
4					
5					

Post condition(s) for Test:

Winner is announced

Project Name: Project 1: Voting System		Team# 20
Test Stage: Unit <u> X </u> System <u> </u>		Test Date: 03/14/2021
Test Case ID#: test_OPLElection_1		Name(s) of Testers: Alex, Nikhil, Peter, Andrew
Test Description: Creates an OPL Election object to run on a ballot csv file		
Automated: yes <u> X </u> no <u> </u>		Indicate where are you storing the tests (what file) and the name of the method/functions being used. OPLElection()
Results: Pass <u> </u> Fail <u> X </u>		[RUN] fixture_OPLBallot.OPLBallotCSVParse_1choice segmentation fault
Preconditions for Test: OPL election is selected		

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create OPL Election	OPLElection()	OPL Election has been created		
2	Run overall structure of OPLElection	run()	Ballot file has been read and results have been compiled and outputted	Segmentation fault!	
3					
4					

Post condition(s) for Test:

All votes have been recorded and counted