

Project Name: Project 1: Voting System
Team# 02

Test Stage: Unit x System

Test Date: 3/25/2024

Test Case ID#: RD_OPL_GSA_1A

Name(s) of Testers: Bethany Freeman

Test Description: Ensuring getSeatAllocation()
function works as intended

**Indicate where are you storing the tests (what file)
and the name of the method/functions being used.**
Project1/src/test/java/ResultsDataOPLTest.java

Automated: Yes X No

Results: Pass X Fail

Preconditions for Test: A ResultsDataOPL object has been created

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create a new ArrayList seatAlloc	seatAlloc = new ArrayList<>();	Object was created successfully	Object was created successfully	
2	Add items to seatAlloc	seatAlloc.add(new ArrayList<>(Arrays.asList("Dem", new int[] { 2, 0 }))); seatAlloc.add(new ArrayList<>(Arrays.asList("Rep", new int[] { 0, 1 })));	Items added successfully	Items added successfully	
3	Assert that the ResultDataOPL object 'test' returns the proper seatAllocation when using the getSeatAllocation when compared to seatAlloc	assertEquals(seatAlloc, test.getSeatAllocation());	True	True	

Post condition(s) for Test:

getSeatAllocation() returns an ArrayList equal to seatAlloc

Project Name: Project 1: Voting System
Team# 02

Test Stage: Unit x System

Test Date: 3/25/2024

Test Case ID#: RD_OPL_GRV_2A

Name(s) of Testers: Bethany Freeman

Test Description: Ensuring getRemainingVotes()
function works as intended

**Indicate where are you storing the tests (what file)
and the name of the method/functions being used.**
Project1/src/test/java/ResultsDataOPLTest.java

Automated: Yes X No

Results: Pass X Fail

Preconditions for Test: A ResultsDataOPL object has been created

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create a new ArrayList remainVotes	remainVotes = new ArrayList<>();	Object was created successfully	Object was created successfully	
2	Add items to remainVotes	remainVotes.add(new ArrayList<>(Arrays.asList("Dem", 84))); remainVotes.add(new ArrayList<>(Arrays.asList("Rep", 250)));	Items added successfully	Items added successfully	
3	Assert that the ResultDataOPL object 'test' returns the proper seatAllocation when using the getSeatAllocation when compared to remainVotes	assertEquals(remainVotes, test.getRemainingVotes());	True	True	

Post condition(s) for Test:

getRemainingVotes() returns an ArrayList equal to remainVotes

Project Name: Project 1: Voting System
Team# 02

Test Stage: Unit x System

Test Date: 3/25/2024

Test Case ID#: RD_OPL_GFWO_3A

Name(s) of Testers: Bethany Freeman

Test Description: Ensuring getFinalWinOrder()
function works as intended

**Indicate where are you storing the tests (what file)
and the name of the method/functions being used.**
Project1/src/test/java/ResultsDataOPLTest.java

Automated: Yes X No

Results: Pass X Fail

Preconditions for Test: A ResultsDataOPL object has been created

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create a new ArrayList finalWinOrder	seatAlloc = new ArrayList<>();	Object was created successfully	Object was created successfully	
2	Add items to finalWinOrder	finalWinOrder.add(new ArrayList<>(Arrays.asList("Dem", "Sarah", 1))); finalWinOrder.add(new ArrayList<>(Arrays.asList("Dem", "Bob", 2))); finalWinOrder.add(new ArrayList<>(Arrays.asList("Rep", "Craig", 3)));	Items added successfully	Items added successfully	
3	Assert that the ResultDataOPL object 'test' returns the proper seatAllocation when using the getSeatAllocation when compared to finalWinOrder	assertEquals(finalWinOrder, test.getFinalWinOrder());	True	True	

Post condition(s) for Test:

getFinalWinOrder() returns an ArrayList equal to finalWinOrder

Project Name: Project 1: Voting System
Team# 02

Test Stage: Unit x System

Test Date: 3/25/2024

Test Case ID#: RD_OPL_GPWO_4A

Name(s) of Testers: Bethany Freeman

Test Description: Ensuring getPartyWinOrder()
function works as intended

**Indicate where are you storing the tests (what file)
and the name of the method/functions being used.**
Project1/src/test/java/ResultsDataOPLTest.java

Automated: Yes X No

Results: Pass X Fail

Preconditions for Test: A ResultsDataOPL object has been created

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create a new ArrayList partyWinOrder	partyWinOrder = new ArrayList<>(Arrays.asList("Dem", "Dem", "Rep"));	Object was created successfully	Object was created successfully	
3	Assert that the ResultDataOPL object 'test' returns the proper seatAllocation when using the getSeatAllocation when compared to partyWinOrder	assertEquals(remainVotes, test.getPartyWinOrder());	True	True	

Post condition(s) for Test:

getPartyWinOrder() returns an ArrayList equal to partyWinOrder

Project Name: Project 1: Voting System
Team# 02

Test Stage: Unit x System

Test Date: 3/25/2024

Test Case ID#: RD_OPL_CWO_5A

Name(s) of Testers: Bethany Freeman

Test Description: Ensuring computeWinOrder()
function works as intended when the partyWinOrder
is empty

**Indicate where are you storing the tests (what file)
and the name of the method/functions being used.**
Project1/src/test/java/ResultsDataOPLTest.java

Automated: Yes X No

Results: Pass X Fail

Preconditions for Test: A FileData object has been created

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create a new ArrayList partyWinOrder	partyWinOrder = new ArrayList<>();	Object was created successfully	Object was created successfully	
2	Create new ArrayList finalWinOrder	ArrayList<ArrayList<Object>> finalWinOrder = new ArrayList<>();	Object was created successfully	Object was created successfully	
3	Create a new ArrayList seatAlloc	seatAlloc = new ArrayList<>();	Object was created successfully	Object was created successfully	
4	Add items to seatAlloc	seatAlloc.add(new ArrayList<>(Arrays.asList("Dem", new int[] { 2, 0 }))); seatAlloc.add(new ArrayList<>(Arrays.asList("Rep", new int[] { 0, 1 })));	Items added successfully	Items added successfully	

5	Create a new ArrayList remainVotes	remainVotes = new ArrayList<>();	Object was created successfully	Object was created successfully	
6	Add items to remainVotes	remainVotes.add(new ArrayList<>(Arrays.asList("Dem", 84))); remainVotes.add(new ArrayList<>(Arrays.asList("Rep", 250)));	Items added successfully	Items added successfully	
7	Initialize a new ResultsDataOPL object	test = new ResultsDataOPL(seatAlloc, remainVotes, partyWinOrder, testFile);	Object was created successfully	Object was created successfully	
7	Assert that the ResultDataOPL object 'test' used the computeWinOrder() method to properly instantiate the finalWinOrder field	assertEquals(finalWinOrder, test.getFinalWinOrder());	True	True	

Post condition(s) for Test:

computeWinOrder() created an ArrayList which was equivalent to finalWinOrder

Project Name: Project 1: Voting System
Team# 02

Test Stage: Unit x System

Test Date: 3/25/2024

Test Case ID#: RD_OPL_CWO_5B

Name(s) of Testers: Bethany Freeman

Test Description: Ensure an IOException is thrown when one party has more votes than it's supposed to

Indicate where are you storing the tests (what file) and the name of the method/functions being used.
Project1/src/test/java/ResultsDataOPLTest.java

Automated: Yes X No

Results: Pass X Fail

Preconditions for Test: A FileData object has been created, testFile

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
2	Create a new ArrayList seatAlloc	seatAlloc = new ArrayList<>();	Object was created successfully	Object was created successfully	
3	Add items to seatAlloc	seatAlloc.add(new ArrayList<>(Arrays.asList("Dem", new int[] { 2, 0 }))); seatAlloc.add(new ArrayList<>(Arrays.asList("Rep", new int[] { 0, 1 })));	Items added successfully	Items added successfully	
4	Create a new ArrayList remainVotes	remainVotes = new ArrayList<>();	Object was created successfully	Object was created successfully	
5	Add items to remainVotes	remainVotes.add(new ArrayList<>(Arrays.asList("Dem", 84))); remainVotes.add(new ArrayList<>(Arrays.asList("Rep", 250)));	Items added successfully	Items added successfully	
6	Initialize a new ResultsDataOPL object and assert that an IOException is thrown	assertThrows(IOException.class, () -> test = new ResultsDataOPL(seatAlloc, remainVotes, new ArrayList<>(Arrays.asList("Pluto", "Pluto", "Pluto")), testFile));	IOException thrown	IOException thrown	

·Post condition(s) for Test:

computeWinOrder() throws an IOException when trying to create a ResultsDataOPL file with an invalid win order

Project Name: Project 1: Voting System
Team# 02

Test Stage: Unit x System

Test Date: 3/25/2024

Test Case ID#: RD_OPL_CWO_5C

Name(s) of Testers: Bethany Freeman

Test Description: Ensure computeWinOrder() works properly under normal circumstances

Indicate where are you storing the tests (what file) and the name of the method/functions being used.
Project1/src/test/java/ResultsDataOPLTest.java

Automated: Yes X No

Results: Pass X Fail

Preconditions for Test: A FileData object has been created, testFile

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Create a new ArrayList winOrder	ArrayList<String> winOrder = new ArrayList<>(Arrays.asList("Pluto", "Pluto", "Green"));	Object was created successfully	Object was created successfully	
2	Create a new ArrayList expected	ArrayList<ArrayList <Object>> expected = new ArrayList<>();	Object was created successfully	Object was created successfully	
3	Add items to expected	expected.add(new ArrayList<>(Arrays.asList("Pluto", "Becky", 1, 20105))); expected.add(new ArrayList<>(Arrays.asList("Pluto", "Mariah", 2, 19943))); expected.add(new ArrayList<>(Arrays.asList("Green", "Jonah", 3, 19943)));	Items added successfully	Items added successfully	
4	Create a new ArrayList seatAlloc	seatAlloc = new ArrayList<>();	Object was created successfully	Object was created successfully	
5	Add items to seatAlloc	seatAlloc.add(new ArrayList<>(Arrays.asList("Dem", new int[] { 2, 0 }))); seatAlloc.add(new ArrayList<>(Arrays.asList("Rep", new int[] { 0, 1 })));	Items added successfully	Items added successfully	

6	Create a new ArrayList remainVotes	remainVotes = new ArrayList<>();	Object was created successfully	Object was created successfully	
7	Add items to remainVotes	remainVotes.add(new ArrayList<>(Arrays.asList("Dem", 84))); remainVotes.add(new ArrayList<>(Arrays.asList("Rep", 250)));	Items added successfully	Items added successfully	
8	Initialize a new ResultsDataOPL object 'test'	test = new ResultsDataOPL(seatAlloc, remainVotes, winOrder, testFile);	Object was created successfully	Object was created successfully	
9	Assert that the finalWinOrder ArrayList is equal to test.getFinalWinOrder()	assertEquals(expected, test.getFinalWinOrder());	True	True	

• **Post condition(s) for Test:**

test.getFinalWinOrder() is equivalent to expected

Project Name: Project 1: Voting System
Team# 02

Test Stage: Unit x System

Test Date: 3/25/2024

Test Case ID#: RD_OPL_GTS_6A

Name(s) of Testers: Bethany Freeman

Test Description: Ensuring the ResultsDataOPL
toString creates the intended string

**Indicate where are you storing the tests (what file)
and the name of the method/functions being used.**
Project1/src/test/java/ResultsDataOPLTest.java

Automated: Yes X No

Results: Pass X Fail

Preconditions for Test: A ResultsDataOPL object has been created, as well as a StringBuilder object
which represents the expected output

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Assert that the toString of the created ResultsDataOPL object outputs the same string as the output string	assertEquals(output.toString(), test.toString());	True	True	

Post condition(s) for Test:

toString() of the ResultsDataOPL object outputs the proper string