Project Name: Project 1: Voting System **Team# 09**

Test Stage: Unit ✔ System __ **Test Date:** 21 March 2018

Test Case ID#: ElectionTest_constructor_01 **Name(s) of Testers:** Q Bayo

Test Description: To test the default values in the election object constructor and the getters/setters to private data of said object.

The unit tests are in the unittest.cc in the testing folder. Once user runs make, a directory is created in the testing folder. Traverse the directory (/build/bin) and execute the executable (unittest).

Automated: yes✔ no ___

Results: Pass ✔ Fail___

Preconditions for Test:

1. default values are set in the constructor
    a. if the default values in the constructor are altered, the unit test must be accommodated.
    b. expected values are fixed constants.
    c. to pass the test, these changes must be accounted for.

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

1	Instantiate the class object	default values of the default constructor in election.cc	num_candidate_ = 0;
			num_seats_ = 0;
			num_ballots_ = 0;
			voting_method_ = 0;
				num_candidate_ = 0;
				num_seats_ = 0;
				num_ballots_ = 0;
				voting_method_ = 0;	

2	calling object's setters	num_candidate_ = 0;
		num_seats_ = 0;
		num_ballots_ = 3;
		voting_method_ = 0;
			num_candidate_ = 0;
			num_seats_ = 0;
			num_ballots_ = 3;
			voting_method_ = 0;
				num_candidate_ = 0;
				num_seats_ = 0;
				num_ballots_ = 3;
				voting_method_ = 0;	

Post condition(s) for Test: In its current state, the object, in this case, election, now has been assigned the above values (2). However, after testing, the object is destroyed via destructor and therefore, the system state remains the same.

Project Name: Project 1: Voting System **Team# 09**

Test Stage: Unit ✔ System __ **Test Date: 21 March 2018**

Test Case ID#: ElectionTest_parseInput_02 **Name(s) of Testers:** Q Bayo

Test Description: To test the parseInput function

The unit tests are in the unittest.cc in the testing folder. Once user runs make, a directory is created in the testing folder. Traverse the directory (/build/bin) and execute the executable (unittest).

Automated: yes✔ no ___

Results: Pass ✔ Fail___

Preconditions for Test:

1. file "input.txt" exists

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

1	Instantiate the class object	default values of the default constructor in election.cc	num_candidate_ = 0;
			num_seats_ = 0;
			num_ballots_ = 0;
			voting_method_ = 0;
				num_candidate_ = 0;
				num_seats_ = 0;
				num_ballots_ = 0;
				voting_method_ = 0;	

2	Test parseInput	input.csv	num_candidate_ = 6; num_ballots_ = 6;	num_candidate_ = 6; num_ballots_ = 6;	

Post condition(s) for Test: After testing, the object is destroyed via destructor and therefore, the system state remains the same.

Project Name: Project 1: Voting System **Team# 09**

Test Stage: Unit ✔ System __ **Test Date: 21 March 2018**

Test Case ID#: ElectionTest_toString_for_election_03 **Name(s) of Testers:** Q Bayo

Test Description: to test the string representation of an election object

The unit tests are in the unittest.cc in the testing folder. Once user runs make, a directory is created in the testing folder. Traverse the directory (/build/bin) and execute the executable (unittest).

Automated: yes✔ no ___

Results: Pass ✔ Fail___

Preconditions for Test:

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	instantiate election object	default values of the default constructor in election.cc	num_candidates_ = 0 num_ballots_ = 0	num_candidates_ = 0 num_ballots_ = 0	
2	call the toString() method	empty string	empty string	empty string	

Post condition(s) for Test: In its current state, the object, in this case, election, now has been assigned the above values (2). However, after testing, the object is destroyed via destructor and therefore, the system state remains the same.