Project Name: Project 2: Voting System **Team# 09**

Test Stage: Unit ✔ System __ **Test Date:** April 22 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 2: Voting System **Team# 09**

Test Stage: Unit ✔ System __ **Test Date: April 22 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: The file "input.txt" was read and the variables num_candidate and num_ballots are set to the number inside the file.

Project Name: Project 2: Voting System **Team# 09**

Test Stage: Unit ✔ System __ **Test Date: April 22 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 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	call the toString() method	empty string	empty string	empty string	

Post condition(s) for Test: In its current state, the object, in this case, the one returned by toString(), now has been assigned the above value(the empty string). However, after testing, the object is destroyed via destructor and therefore, the system state remains the same.

Project Name: Project 2: Voting System **Team# 09**

Test Stage: Unit ✔ System __ **Test Date: April 19 2018**

Test Case ID#: ElectionTest_Election_Plurality_04 **Name(s) of Testers:** Shalom Nguyen

Test Description: to test plurality

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:

A candidate list and a ballot list must exist

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	call the setNum_candidates	num_candidate_ = 2;	num_candidate_ = 2;	num_candidate_ = 2;	
3	call the setNum_ballots	num_ballots_ = 1;	num_ballots_ = 1;	num_ballots_ = 1;	
4	call the setNum_seats	num_seats_ = 0;	num_seats_ = 0;	num_seats_ = 0;	

5	call setBallot_list	Ballot* bal_lst = new Ballot[1];
		int* tmp = new int[2];
		tmp[0] = 0;
		tmp[1] = 1;
		bal_lst[0].setList_of_ranks(tmp);
		bal_lst[0].setNum_candidates(2);
			tmp[0] = 0;
			tmp[1] = 1;
			bal_lst[0].setList_of_ranks(tmp);
			bal_lst[0].setNum_candidates(2);
				tmp[0] = 0;
				tmp[1] = 1;
				bal_lst[0].setList_of_ranks(tmp);
				bal_lst[0].setNum_candidates(2);
					precondition

6	call setCandidates_list	Candidate* cand_lst = new Candidate[2];
		cand_lst[0].setCandidate_name("A");
		cand_lst[1].setCandidate_name("B");
			cand_lst[0].setCandidate_name("A");
			cand_lst[1].setCandidate_name("B");
				cand_lst[0].setCandidate_name("A");
				cand_lst[1].setCandidate_name("B");
					precondition

7	test runPlurality	cand_lst, bal_lst	B is in the Winner_list A is in the Alternate_list	B is in the Winner_list A is in the Alternate_list	Candidate* winners = election.getWinner_list(); Candidate* losers = election.getAlternate_list();

Post condition(s) for Test: Candidate B is now put in Winner_list and candidate A is put in Alternate_list after the function runPlurality.

Project Name: Project 2: Voting System — Team# 09

Test Stage: Unit ✔ System __ **Test Date: April 19 2018**

Test Case ID#: ElectionTest_Election_Droop_05 **Name(s) of Testers:** Shalom Nguyen

Test Description: to test Droop

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:
A candidate list and a ballot list must exist

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	call the setNum_candidates	num_candidate_ = 2;	num_candidate_ = 2;	num_candidate_ = 2;	
3	call the setNum_ballots	num_ballots_ = 1;	num_ballots_ = 1;	num_ballots_ = 1;	
4	call the setNum_seats	num_seats_ = 0;	num_seats_ = 0;	num_seats_ = 0;	

5	call setBallot_list	Ballot* bal_lst = new Ballot[1];
		int* tmp = new int[2];
		tmp[0] = 0;
		tmp[1] = 1;
		
		bal_lst[0].setList_of_ranks(tmp);
		
		bal_lst[0].setNum_candidates(2);
			tmp[0] = 0;
			tmp[1] = 1;
			
			bal_lst[0].setList_of_ranks(tmp);
			
			bal_lst[0].setNum_candidates(2);
				tmp[0] = 0;
				tmp[1] = 1;
				bal_lst[0].setList_of_ranks(tmp);
				bal_lst[0].setNum_candidates(2);
					precondition

6	call setCandidates_list	Candidate* cand_lst = new Candidate[2];
		
		cand_lst[0].setCandidate_name("C");
		
		cand_lst[1].setCandidate_name("D");
			cand_lst[0].setCandidate_name("C");
			
			cand_lst[1].setCandidate_name("D");
				cand_lst[0].setCandidate_name("C");
				cand_lst[1].setCandidate_name("D");
					precondition

7	test runPlurality	cand_lst, bal_lst	D is in the Winner_list C is in the Alternate_list	D is in the Winner_list C is in the Alternate_list	Candidate* winners = election.getWinner_list(); Candidate* losers = election.getAlternate_list();

Post condition(s) for Test: Candidate D is in the Winner_list and candidate C is in the alternate_list.

Project Name: Project 2: Voting System **Team# 09**

Test Stage: Unit ✔ System __ **Test Date: April 22 2018**

Test Case ID#: ElectionTest_sortCandidateByVotes_06 **Name(s) of Testers:** Shalom Nguyen

Test Description: to test sortCandidateByVotes

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:

A candidate list must exist : use i.e

```
Candidate* cand_lst = new Candidate[5];
cand_lst[0].setNum_ballots(2);
cand_lst[1].setNum_ballots(3);
cand_lst[2].setNum_ballots(5);
cand_lst[3].setNum_ballots(9);
cand_lst[4].setNum_ballots(7);
```

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	call the setNum_candidates	num_candidate_ = 5;	num_candidate_ = 5;	num_candidate_ = 5;	
3	call sortCandidateByVotes	cand_lst[0] = 9 cand_lst[1] = 7 cand_lst[2] = 5 cand_lst[3] = 3	cand_lst[0] = 9 cand_lst[1] = 7 cand_lst[2] = 5 cand_lst[3] = 3	cand_lst[0] = 9 cand_lst[1] = 7 cand_lst[2] = 5 cand_lst[3] = 3	see precondition test

		cand_lst[4] = 2	cand_lst[4] = 2	cand_lst[4] = 2	

Post condition(s) for Test: The candidate array is sorted by the number of votes - the candidate with the most number of votes are at index 0, and the least are at index n.