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

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

Test Case ID#: BT_01 **Name(s) of Testers:** Jay Uppaluri

Test Description: Testing the default values in the ballot object constructor and the getters/setters.

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 unittest must be compiled again
    b. expected values are fixed constants

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Instantiate the class object	Default values of the default constructor in ballot.cc	ballot_id = 5 num_candidates = 0 list_of_names_ = NULL list_of_ranks_ = NULL	ballot_id = 5 num_candidates = 0 list_of_names_ = NULL list_of_ranks_ = NULL	
2	Calling the object's setters	ballot_id = 10 num_candidates = 5 list_of_names_ = {"john", "mary", "bromeo"} list_of_ranks_ = {1, 2, 3, 4}	ballot_id = 10 num_candidates = 5 list_of_names_ = {"john", "mary", "bromeo"} list_of_ranks_ = {1, 2, 3, 4}	ballot_id = 10 num_candidates = 5 list_of_names_ = {"john", "mary", "bromeo"} list_of_ranks_ = {1, 2, 3, 4}	

Post condition(s) for Test:

Ballot now has been assigned the above values of step #1 due to the setters.

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

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

Test Case ID#: BT_02 **Name(s) of Testers:** Jay Uppaluri

Test Description: Testing the ballot.toString 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. ballot_id exists, num_candidates exists, and list_of_ranks exists

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Instantiate the class object	Default values of the default constructor in ballot.cc	ballot_id = 5 num_candidates = 0 list_of_names_ = NULL list_of_ranks_ = NULL	ballot_id = 5 num_candidates = 0 list_of_names_ = NULL list_of_ranks_ = NULL	
2	Calling the object's setters	ballot_id = 10 num_candidates = 4 ranks = {1, 2, 3, 4}	ballot_id = 10 num_candidates = 4 ranks = {1, 2, 3, 4}	ballot_id = 10 num_candidates = 4 ranks = {1, 2, 3, 4}	
3	Test toString	ballot_id = 10 num_candidates = 4	ballot.toString() = "ID: 10 Count: 4\n1 2 3 4"	ballot.toString() = "ID: 10 Count: 4\n1 2 3 4"	

		ranks = {1, 2, 3, 4}			

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#: BT_03 **Name(s) of Testers:** Jay Uppaluri

Test Description: Testing the ballot.findCandidate 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:

2. num_candidates exists, ranks is an array and exists.

Step #	Test Step Description	Test Data	Expected Result	Actual Result	Notes
1	Instantiate the class object	Default values of the default constructor in ballot.cc	ballot_id = 5 num_candidates = 0 list_of_names_ = NULL list_of_ranks_ = NULL	ballot_id = 5 num_candidates = 0 list_of_names_ = NULL list_of_ranks_ = NULL	

2	Calling the object's setters	num_candidates = 5 ranks = {1, 2, 3, 4}	num_candidates = 5 ranks = {1, 2, 3, 4}	num_candidates = 5 ranks = {1, 2, 3, 4}	
3	Test findCandidate	num_candidates = 5 ranks = {1, 2, 3, 4}	ballot.findCandidate(1) = 0;	ballot.findCandidate(1) = 0;	

Post condition(s) for Test:

After testing, the object is destroyed via destructor and therefore, the system state remains the same. the findCandidate function returns the rank of the