

I want the OPL and IRV algorithms to work fully.

Acceptance Criteria:  
-Both OPL and IRV algorithms output the correct winner and an audit file is created showing how the election progressed.  
-OPL and IRV work as described in project 1

EFFORT: Small

Fix bug caught by TA in IRV election

Update election class equals method

Report any new bugs in the Bug List

Write code for IRV election

Document the code and generate the javadocs

Unit testing for both the algorithms

Complete the testing logs for each test

System test to check both algorithms with input file

Write code for OPL Election

I want invalidated ballots to be removed from the IRV election when run by the software system with a file created to store invalid ballots. At least half of candidates need to be ranked for a ballot to be valid.

Acceptance Criteria:  
-A invalid ballots file is created with all invalid ballots -invalid ballots are successfully determined and removed from the election.

EFFORT: Medium

Check for and remove any bad ballots in code

Document the code

System test to see if the bad ballots are removed from file

Unit test to check removed ballots

I want to see a table of how the election progressed each round during an IRV election. Table will be displayed to the screen with important information

Acceptance Criteria:  
-A table is created which shows each round of IRV with the table being displayed to the screen  
-Table contains correct data which accurately reflects the rounds of the IRV election

EFFORT: Large

Cleanup code if required

Document the code

Code using Test Driven Development (TDD) to create a table with all the information

Ensure table is correct by running system test using input file

Code to pad the table to make it look nice

Table can automatically update the number of rounds necessary

I want to input a file as a command line argument when running election

Acceptance Criteria:  
-Can input a file as a command line argument and election will parse it and run successfully  
-Election will parse the file and output correct winner

EFFORT: Small

Test for the number of arguments in the command line

Document the code

Write Code using Test Driven Development (TDD) and pair programming to input file name through command line argument

Ensure both command line and regular input still work

Run system test to ensure command line works

Ensure code only prompts user for filename

Document the code

I want a text prompt for input file to show if no command line arguments are given.

Acceptance Criteria:  
-A prompt is given if no command line argument is given -Prompt will take a file and use that for the election  
-Successfully parses through file and output the winner

EFFORT: Small

Run system tests to ensure code pulls all data from file

Write unit tests with mock file to check readInput

System test of GUI to ensure it works properly

Verify if the code produces an audit file

Document the code

Write code using JFrame to have gui input

Import JFrame into project to use

Unit testing for edge cases

Update the project to use JUnit 5 for unit tests

I want a GUI to input the file. A window should appear for user to type in file.

Acceptance Criteria:  
-A GUI pops up with a place to input a file  
-Algorithm will successfully take that file as input and parse it

EFFORT: Large