

## Project One User Manual

Welcome to using the Voting Machine Simulator. This software allows the user to utilize a heavily simplified voting mechanism, allowing one to cast votes for a certain race and retrieve the results afterwards.

The program additionally provides the user with a GUI (Graphical User Interface) that facilitates usage of the software.

To begin, the user must provide a correctly formatted input file, describing the name of the election/ position in office, number of total candidates, and the names and affiliation of each candidate involved. Specifically,

- The first line must denote the name of the election/ position in office
- The second line must denote the number of candidates running
- The subsequent lines will each correspond to a candidate in the following format:  
"Name;Affiliation"

For example,

```
James Beard Award
3
Nina Compton;Compere Lapin
Alon Shaya;Saba
Emeril Lagasse;Emeril's
```

The program will then read the input file, perform calculations on the data and publish the election results to a given output file in a formatted way.

For example,

```
RESULTS - James Beard Award
-----
Nina Compton - Compere Lapin      102
Alon Shaya - Saba                  105
Emeril Lagasse - Emeril's         97

WINNER: Alon Shaya - Saba
```

Below is a description of the implemented classes and corresponding methods involved

### **Candidate.java**

Candidate

Class constructor used for generating Candidate object.

getName

Method used for retrieving Candidate object's name.

getAffiliation

Method used for retrieving Candidate object's affiliation.

getVoteCount

Method used for retrieving Candidate object's vote count.

tallyVote

Method used for incrementing Candidate object's vote count by 1.

toString

Method used for generating a string representation of Candidate object.

compareCandidate

Method used for comparing the vote count of two Candidate objects.

### **Ballot.java**

Ballot

Class constructor used for generating Ballot object.

getOfficeName

Method used for retrieving Ballot object's office name.

addCandidate

Method used for adding a Candidate object to the Ballot object's array of candidates.

getCandidates

Method used for retrieving the list of Candidate objects in Ballot object.

### **BallotReader.java**

BallotReader

Class constructor used for generating a BallotReader object.

readBallot

Reads a formatted input file describing a political race and retrieves the name of office, candidates running and their affiliations. Returns a Ballot object completed with corresponding information.

## **ResultWriter.java**

### ResultWriter

Class constructor used for generation a ResultWriter object.

### writeResults

Given a ballot and file, the method writes the results of the election in a formatted way onto the file.