ABSTRACT

Our application is a voting machine, which helps all kinds of organizations to do public voting for any awards or elections. This program needs voters to vote on this one by one, and it will collect every voters' result and do the counting, displaying the candidate who has the maximum votes. More amazingly, this application also includes a further extension. It allows the host of the event to add more options, adding more potential candidates; therefore, the voters will be able to have more options to vote. Here, we will provide introduction and the instructions of this application.

INTRODUCTION

Input

The program reads ballot information from an input file, which is the initial file along with some number of options people can vote for, an example of which follows:

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

The details of the input formats are:

- First line being set to be the award or election which people need voting to decide

- Second line showing the number of options that voters may have

- Next three lines specifying a candidate each, with the candidate's name appearing on the line first, followed by a semicolon, and followed by the candidate's party affiliation

Here, the example shows that some unknown organization needs voting to decide the winner for the James Beard Award. In this case, voters have three options, including Nina Compton from Compere Lapin, Alon Shaya from Saba, and Emeril Lagasse from Emeril’s.

We assume the input uses correct format like the one we discussed above, which also needs to be TXT file. However, if the format of input is not correct, the program will misbehave or even crash. In other word, the program will only test your program with valid input files.

Output

After voting, the program will implement PrintStream to display the candidate who has the maximum number of votes in the window, which becomes the winner. Besides, the addition candidates which are added from the host will also be included into the comparison, and by comparing the amount of votes of each candidates, displaying the winner in the end. Meanwhile, a nicely formatted output file that indicates the result will be created and the application will provide an option available for saving the file or not, after displaying the winner in the window. An example of the output format follows:

```
RESULTS - James Beard Award
--------------------------
Nina Compton - Compere Lapin      5
Alon Shaya – Saba                 7
Emeril Lagasse - Emeril’s         2
Adam Smith - Nobel                4
 Steve White - new era             3

WINNER: Alon Shaya - Saba
```

The details of the output format are:

- First line beginning with the word "RESULTS", followed by a space, a dash, another space, and then name of the award or election that users voted for

- A line of dashes, where there exactly as many dashes as there are characters on the first line

- Next five lines showing each candidates on the ballot including two candidates who were added before the vote with each candidate appearing on a separate line:
  - The name of the candidate, a space, a dash, a space, and the candidate's party affiliation, calling all this the candidates' tags
  - The number of votes received by that candidate
  - The candidate names should be left-justified in the first column of the output.
  - The numbers of votes should be right-justified in the last column of the output.

- A blank line appearing after all of the candidates are listed with their number of votes

- A line that indicates the winner, with the word "WINNER", a colon, a space, and the winning candidate's tag

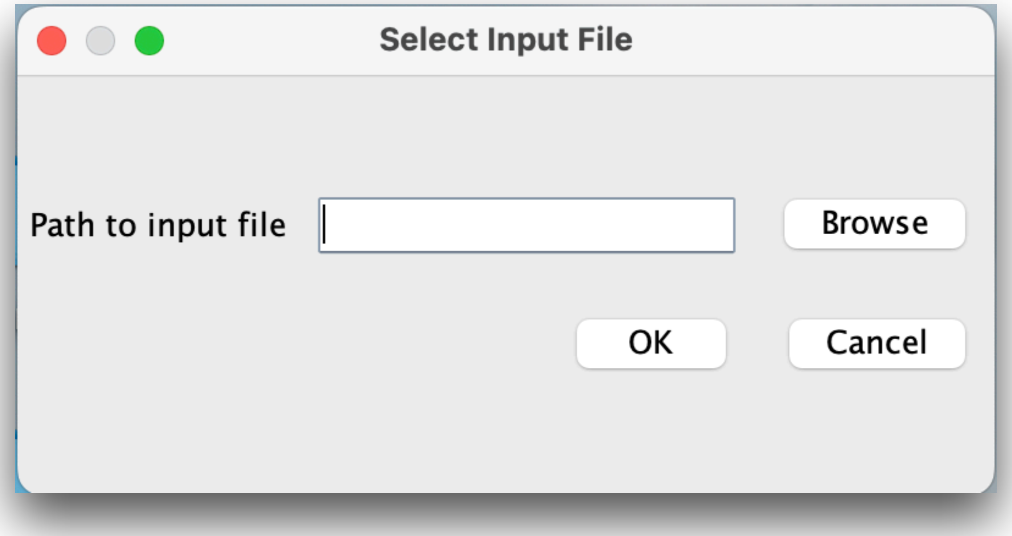
The example shows that Nina Compton from Compere Lapin gets 5 votes, Alon Shaya from Saba gets 7 votes, Emeril Lagasse from Emeril’s gets 2 votes, Adam Smith from Nobel gets 4 votes, and Steve White from new era gets 3 votes. The overall winner is Alon Shaya from Saba, since she gets the most votes.

INSTRUCTIONS

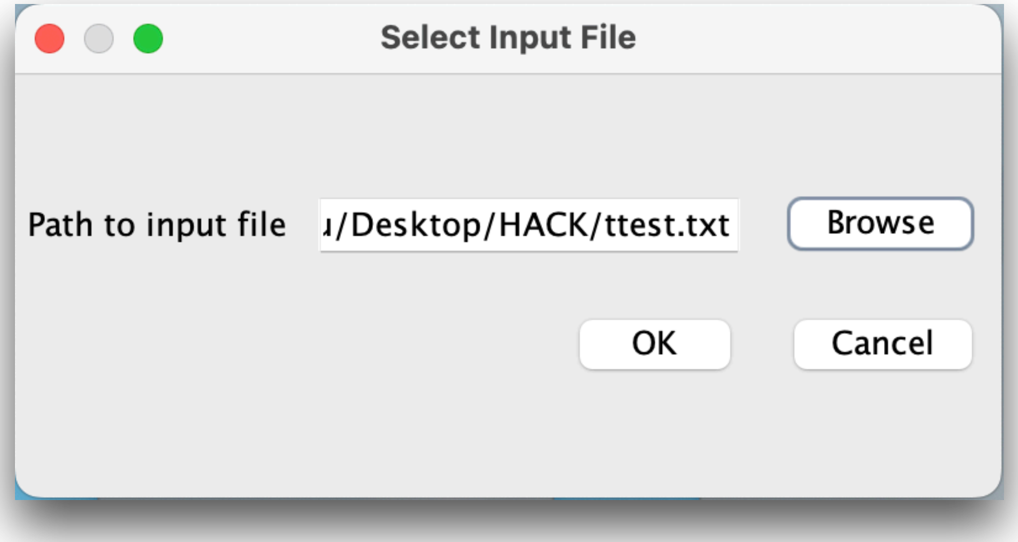
This application involves many windows along with buttons, making the vote process easy and convenient. Besides, it also makes the human error interpretable by jumping out warning windows to help you solve the error, making this voting system easier to achieve.

The overall process: (using the same example in the Introduction part)

- Open the application and getting the following window

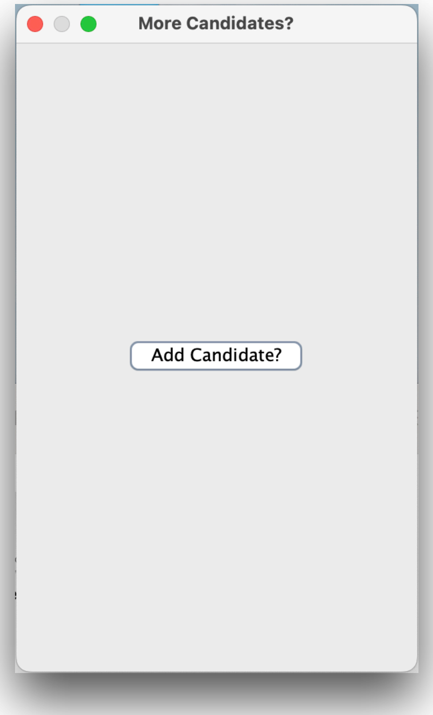


- Upload the input file using the correct format

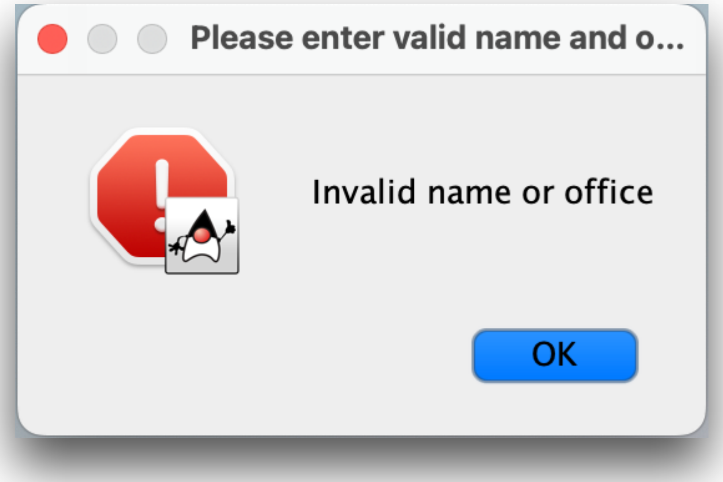


ı/Desktop/HACK/ttest.txt

- Then choosing if you want to add another candidate: if it is false, close the current window; if it is true, click the button “Add Candidate?”:



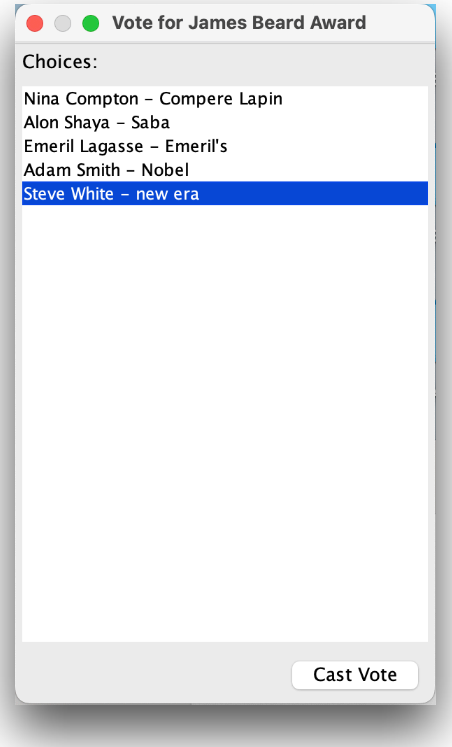
- Input the new candidate's name and his party affiliation in the next two following windows
- If you miss either one of these two inputs, a error will come out


Please enter valid name and o...

Invalid name or office

- Close the window of adding more candidates

- Then start voting: choose one of the candidates and click on the "Cast Vote" button



Vote for James Beard Award

Choices:

Nina Compton – Compere Lapin
Alon Shaya – Saba
Emeril Lagasse – Emeril's
Adam Smith – Nobel
Steve White – new era

- After finishing the voting, close the window and there will be a new window jump out, displaying the result; if there is a draw, it will display “NO WINNER”

- At last, coming with a window of choosing to save the file(result) or not and choosing the path

CONCLUSION

Our application created as vote machine can be the substitution of the traditional voting, but it appears to be the improved application since it is more convenient and easy to achieve. Besides, during the voting, participants are not able to know others’ decisions and also no one else is able to know his decision, which ensures the privacy. One potential extension of our application is to make it be able for voters to vote on different devices and gather all the information around to display the conclusion.