CS 161 Program Five

9 Points

Purpose

This assignment introduces the use of arrays (lists) in programming. After finishing this program you will know how to use lists to store information and how to use lists to accumulate sets of values. You will also gain more experience in breaking down a problem solution into functions, and more experience debugging code.

Scenario

In recent years, paperless voting machines using computers with touch-screen interfaces have been in the news quite a bit. This program is a grossly oversimplified computerized voting system. The real McCoy would of course have many more places for things to go wrong!

This program will operate in three phases. The first phase simulates the election workers setting up the machine. The program will ask for the number of candidates, and will then ask the worker to enter each candidate's name and then party. This information will be stored in two lists, and then the second phase will begin.

The second phase simulates the voting period. The program will repeatedly list a numbered slate of candidates and their parties, and invite voters to select a candidate by entering a number. *If the voter enters a number not in the list, the program ignores the entry*. Each valid vote is tallied in an integer list that corresponds to the list of candidate names. When an election worker enters a negative number for the vote, the election is over and the program enters phase three.

The last phase is the reporting phase. The program prints a report showing each candidate's name, party, number of votes, and percentage of total votes. It then prints a histogram showing candidate names and a bar representing the percentage of vote earned by that candidate.

Program Requirements

As with previous assignments, this one has different stage of completion worth various amounts of credit. Start by solving the simplest version, and then save a backup and work on the more complex versions. Turn in the best *complete* version you finish.

For any version, your program should have the correct header and report format, and should report for all candidates the correct number of votes.

Candidate names are one to twelve characters.

Parties will be no more than three characters (as in "R" for "Republican," "IND" for "Independent" and "N" for "Nutso").

The prompt printed before each input request shows "1" for the first candidate number.

The program is divided into logical functions.

The program **and** each function should be documented. Function documentation should describe what the function does, identify any arguments and the value returned (if any). Inadequate or sloppy documentation will result in a loss of up to 1/3 credit.

Basic Version—Complete for 5 Points Maximum

This version does not compute percentages or create a histogram. It produces a neatly formatted report showing the candidate names, their parties, and the number of votes they received.

Intermediate Version—Complete for 7 Points Maximum

In addition to satisfying the requirements for the basic version, this version stores the party for each candidate (in a separate list), adds the party to the prompt and the report, and accurately computes and reports the percent of the total vote earned by each candidate. For reporting, the percent will be rounded to two decimal places.

Advanced Version—Complete for 9 Points Maximum

For this version, there are additional requirements. The first is that the program produces a histogram that shows for each candidate, the name as a label, a vertical bar, and a row of asterisks (*) with each one representing one percentage point. The second is to make the reports tidy even if some candidate has a short name like "Wu" and another has a long name like "Kulongoski."

Some Tips

To figure out what code to encapsulate in functions, start by making an outline of the general steps of the program. This outline should describe what your main function looks like, and indicate logical chunks for subordinate functions. If a subordinate function seems unduly complex, make a rough outline of what's necessary for it to work, and use this as a guide to subdivide the code into more sub-functions.

The prompt shown to voters in the second phase will be the same for the duration of program execution. It would make sense not to re-construct it each time it is displayed.

Computing percentages accurately requires doing multiplication before division, and including at least one float value in the computation. In the python interactive environment, try these attempts to compute what percent 23 is of 47:

```
>>>23/47*100
>>>23*100/47
>>>23*100.0/47
```

Remember that the **round** function can be used to round a floating point number to a specified number of decimal places by adding a second argument to the call.

To create the histogram, you'll need to come up with an integer value representing the number of half percentage points a percentage represents. If you manually compute this for a few numbers, a simple solution should present itself.

Handling various name lengths requires determining how long the name is and padding it with spaces until a standard length is reached.

Sample Output is on the next page

Phase One Activity

```
Please enter the number of candidates: 5
Please enter the name of candidate 1: Whoosh
Please enter Whoosh's party: REP
Please enter the name of candidate 2: Terry
Please enter Terry's party: DEM
Please enter the name of candidate 3: Li
Please enter Li's party: IND
Please enter the name of candidate 4: Troutfisher
Please enter Troutfisher's party: H2O
Please enter the name of candidate 5: Swindle
Please enter Swindle's party: LBY
```

Phase Two Sample

```
Vote by entering the number of your chosen candidate.

1    Whoosh  (REP)
2    Terry  (DEM)
3    Li  (IND)
4    Troutfisher  (H2O)
5    Swindle  (LBY)

Enter your vote: 1
```

Phase Three Output

```
=============== Election Results ===============

Candidate   Party   Votes     Percent


Whoosh       REP      4       11.43
Terry        DEM      7       20.0
Li           IND      12      34.29
Troutfisher  H2O      7       20.0
Swindle      LBY      5       14.29


======= Histogram  =======


Whoosh       | ***********
Terry        | ********************
Li           | **********************************
Troutfisher  | ********************
Swindle      | **************
```