CSCE 215 – Unix/Linux Fundamentals
Fall 2017 – Assignment 2
10 Points

Due: November 3, 2017 5pm

Overview

This assignment will get you familiar with using Unix tools to manipulate file data. As you may know, the presidential election is a two stage process: each state in the US holds separate elections, deciding electoral votes; then, the electoral votes are used to select the president. As such, strategists focus on elections at the state level, and polls are regularly taken on a per-state basis. The site `electoral-vote.com` regularly collects this data. We have access to the raw spreadsheets from the 2008 General Election in the directory `/usr/local/tmp/csce215/election2008`.

Directory Organization

The data has been organized as follows:

> Under the directory `/usr/local/tmp/csce215/election2008`, there are subdirectories for each month, which contain both `.csv` and `.xls` files. The `.xls` files are Microsoft Excel format, so you won't be able to view the contents without an appropriate application, but the `.csv` files contain the same data in ASCII readable format. The month names are 3-letter long, and each file is of the format `mmmdd.ext`, where `mmm` is the month, `dd` is the day (always two digits), and `ext` is the extension (either `.csv` or `.xls`).

We will only be looking inside the `.csv` files. One such file exists per day, and contains the most recent polls for all states. They consist of three parts:

- A 2 line header, describing the fields.
- 51 lines representing data for each state (plus the District of Columbia).
- The rest of the file is summary information.

The data fields for each state in the second part is comma separated. The names of the fields are listed in the first line, which are:

State: Name of the state the poll was taken in
EV: number of electoral votes the state has
Dem: Percent of voters voting for Obama
Rep: Percent of voters voting for McCain
Ind: Percent of voters voting for Ralph Nader
Date: The date the poll was taken
=10: This column is the same as EV if Obama has at least a 10% lead in the state
5-9: This column is the same as EV if Obama has a 5-9% lead in the state
<5: This column is the same as EV if Obama has less than a 5% lead in the state
Tie: This column is the same as EV if both candidates are tied
<5: This column is the same as EV if McCain has less than a 5% lead in the state
5-9: This column is the same as EV if McCain has a 5-9% lead in the state
=10: This column is the same as EV if McCain has at least a 10% lead in the state
Poll source: The name of the organization that conducted the poll

Before you begin

Before answering the questions, explore the directories and experiment with Unix commands. In particular, familiarize yourself with the commands:

cd ls cat head tail cut sort uniq tr wc find

Questions

Each of these questions should be expressible as a single command **or** pipeline of commands that run from your assignment 2 sub-dir. You should not need multiple lines or semicolons for any question.

Part 1 (1 point each)

Using `find`, write commands that search for files in the following directory:
`/usr/local/tmp/election2008`

1. List all `.csv` files.
2. List `.csv` files in the sub-dir Jul.
3. List all files from the first 9 days of August.
4. List all files from the first 9 days of July and August.
5. List only `.csv` files from before August 10.

Part 2 (5 points)

Using `head` and `tail`, write a command to extract the second section of a file (i.e. the data section). If you are really confused, this info is bullet #2 from the previous page.

Turn this into an executable script called `extractdata` (you do not need to hand this in). Then, using `find` and `extractdata`, write a command to get the second section of all `.csv` files in the month directories, and place the output into a file called `polls.csv`. Be sure to keep this file in your 215 homedir. You will use it again on the next assignment.

Hint: Inside the script don't forget the command line variable $1.
Example: head -52 $1

Submission (use create_assignment)

All commands should be submitted in a single file, named

[yourUserID].assignment2.txt

which is what you will turn in. Before each command, clearly state which question number you are answering. Here is an example:

```
# Question 1:
find .  -name myfile -print

# Question 2:
find .  -name otherfile -print

.
.
.
# Question 6:
find blah blah blah somehow use ./extractdata
```

- Make sure each command has been tested!
- Keep in mind, that all I am asking for is the commands themselves, **NOT** the output of the commands!
- When you have finished, submit the file containing the commands using the departmental dropbox.