Caleb Javier

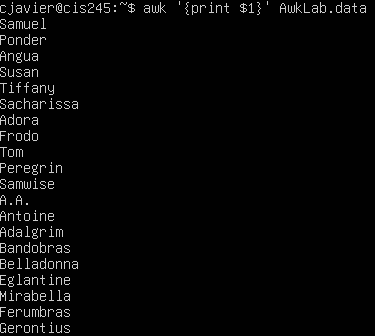
Professor Holden-Gouveia

CIS-245

29 September 2024

Week 4: Awk

1. **Print all the first names.**

****

By using $ followed by a number, the text corresponding to the number and how many delimiters (whitespaces) are present will be printed.

1. **Print phone numbers for Tom and Frodo after their names.**

****

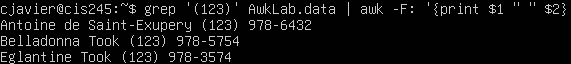
This sets the delimiter to the colon using -F option, and then pipes the command into a grep search for either Tom *or* Frodo. The -E flag in grep allows for the use of extended regex, which is where the OR operator comes from.

1. **Print Peregrin’s full name and phone number area code only.**

****

By again using piping and the default delimiter, Peregrin’s full name can be extracted and the final output cleaned up by substituting the colon with a space. A literal space is placed between the numbered sections to also help with presentation.

1. **Print all phone numbers with the 123 area code along with names.**

****

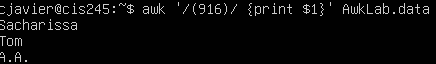
By this time using grep first, all phone numbers with 123 area code can be filtered for.

1. **Print all last names beginning with either a T or D.**
2. **Print all first names containing four or less characters.**

****

By determining a length and setting it as less than or equal to four, all names four characters or less are printed.

1. **Print the first names and area codes of all those with the 916 area code.**

****

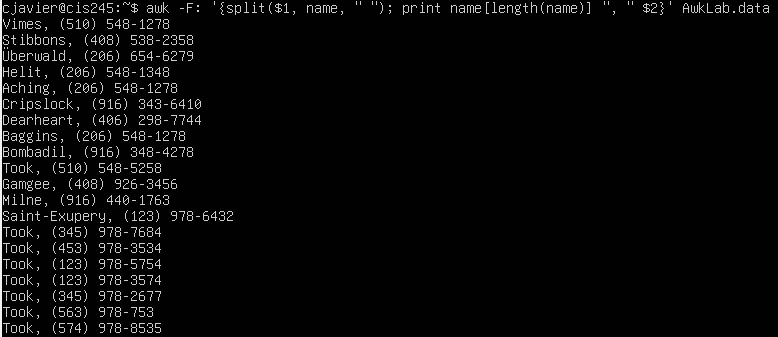
This time, a pattern is selected to be matched where the first field is printed.

1. **Print Sacharissa’s name followed by her campaign contributions.**



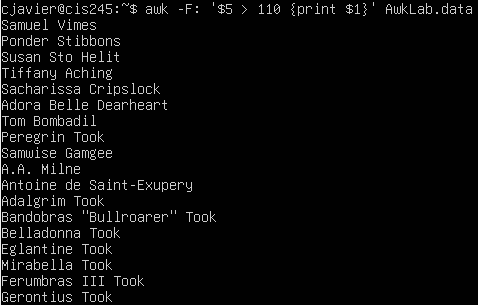
By using a colon as the delimiter and locating her name using a snippet of it, then printing by field, this pattern can be obtained.

1. **Print the last names followed by a comma and the phone number.**



The split command is able to create an array out of the delimited field, where it can be printed as a single element. The “name” is what the array is called, and the literal space in quotes is the new delimiter. The comma and space are printed literally followed by the second field.

1. **Print the first and last names of those who contributed more than $110 in the last month.**

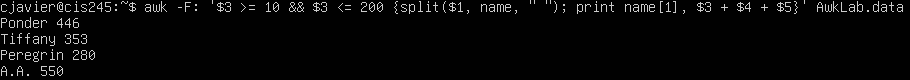
****

By using awk to find where the fifth field (last month’s contribution) is greater than $110, each line corresponding to this condition can be printed.

1. **Print the last names, phone numbers, and first month contribution of those who contributed less than $150 in the first month.**

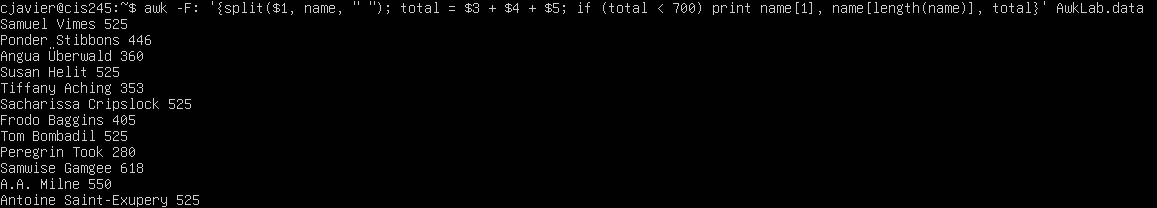
****

1. **Print the first names and contributions of those who contributed between $10 and $200 in the first month.**

****

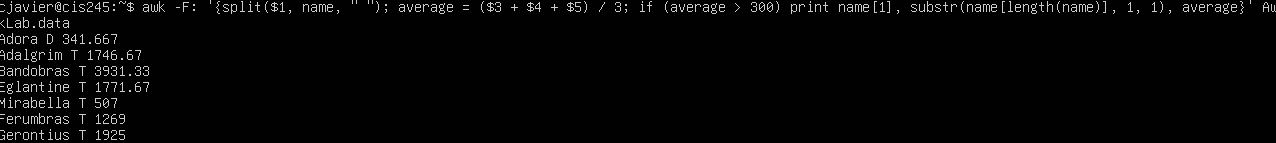
By using the and boolean and creating an expression for the first month contribution, we can utilize the same name split command to find the first names and totals of the lowest first month contributors.

1. **Print the first and last names and total contributions of those who gave less than $700 over the three-month period.**



The total value can be set equal to a variable for which a comparison can be made by using the less than operator, which allows the names who gave less than 700 be printed.

1. **Print the first name, first letter of the last name, and average contribution of those who gave an average of more than $300.**

****

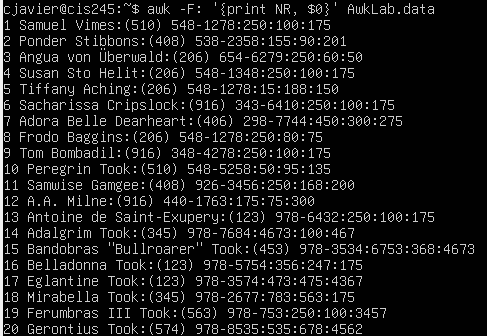
Again a variable is made for the average, which will simply be called average. The substr command is used to simplify finding the first letter of the last name, and the average is used to determine if the total donation is less than 300.

1. **Print the last name and area code of those not in the 916 area code.**

****

The if statement used in this command is able to filter where the pattern is **not** by using the negation match operator.

1. **Print each record preceded by the number of the record.**

****

The NR variable is used to record the number of lines. By printing it before the first field (0) of the entire line, each record can be printed preceded by its entry.

1. **Print the name and total contribution of each person.**

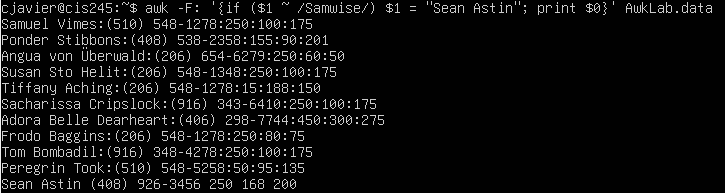
****

1. **Add $10 to Tiffany Aching’s first contribution and print it with her full name.**

****

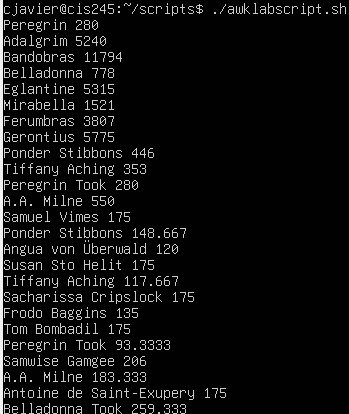
By matching the result with Tiffany’s name and setting the field $3 equal to $3 + 10, 10 can be added to her first month donation.

1. **Change Samwise Gamgee’s name to Sean Astin.**

****

After matching the pattern to Samwise’s name to ensure the right line is being edited, the first field (populated by the name) can be set equal to “Sean Astin”, literal string.

1. **Write an awk script to do the following:**
   1. **Print the first name of all the Tooks followed by their total contribution.**
   2. **Print the full names and contributions of anyone who gave between $10 and $200 in the last month.**
   3. **Print the full names and average contribution of those who gave less than $300 on average.**

****

The script is attached; this is the result of all three parts.

Works Cited

“AWK Command in Unix/Linux with Examples.” *GeeksforGeeks*, GeeksforGeeks, 12 July 2024, www.geeksforgeeks.org/awk-command-unixlinux-examples/.

“Getting Substring with Awk.” *Go to Forum Page*, www.unix.com/shell-programming-and-scripting/132239-getting-substring-awk.html. Accessed 6 Dec. 2024.

*GREP(1) - Linux Manual Page*, man7.org/linux/man-pages/man1/grep.1.html. Accessed 6 Dec. 2024.

hqjmahqjma 4, et al. “How Do I Use Shell Variables in an Awk Script?” *Stack Overflow*, 1 Nov. 1958, stackoverflow.com/questions/19075671/how-do-i-use-shell-variables-in-an-awk-script.

Mohamed SalighMohamed Saligh 12.3k1919 gold badges6767 silver badges8484 bronze badges, et al. “How to Split a Delimited String into an Array in Awk?” *Stack Overflow*, 1 Dec. 1956, stackoverflow.com/questions/8009664/how-to-split-a-delimited-string-into-an-array-in-awk.

projects, Contributors to Wikimedia. “Regular Expressions/POSIX-Extended Regular Expressions.” *Wikibooks, Open Books for an Open World*, Wikimedia Foundation, Inc., 12 May 2011, en.wikibooks.org/wiki/Regular\_Expressions/POSIX-Extended\_Regular\_Expressions.