CIT 371 Lab 11: Regular expressions and egrep continued

This lab can be done with SSH/PuTTY or with the Web Console. See the Student VM Access document for information on accessing your VMs.

Each step of this lab requires that you come up with an egrep command that finds the lines requested. Your answer is that egrep command, not what you found. Log into Coivcenter, start your VM and log in as Student. From ~Student/FILES, issue the following three wget commands (these commands download the given files to the current directory).

* **wget --no-check-certificate www.nku.edu/~foxr/equals.txt**
* **wget --no-check-certificate www.nku.edu/~foxr/names.txt**
* **wget --no-check-certificate www.nku.edu/~foxr/sentences.txt**

1. Using equals.txt
   1. *Find all lines that contain an open parenthesis.*
      1. **Egrep ‘\(‘ equals.txt**
   2. *Find all lines that contain an assignment statement of the form variable = value where value is a numeric value.* 
      1. **Egrep ‘([A-Z]+ \= [0-9])’ equals.txt**
   3. *Find all lines that contain a mathematical equation of the form value operator value = value where operator can be any mathematical operand and value is any numeric value. The equation does not have to be valid (for instance 9 \* 10 = 11).* 
      1. **Egrep ‘[0-9]+ [+\*/-] [0-9]+ = [0-9]+$’ equals.txt**
   4. *Find all lines that are sequences and not equations or assignment statements. A sequence is any collection of letters/digits separated by spaces.*
      1. **egrep ‘^[a-zA-Z0-9]+$’ equals.txt**
2. Using sentences.txt
   1. *Find all sentences that have words where there are two vowels in a row.*
      1. **Egrep ‘[a|e|i|o|u] [a|e|i|o|u]’ sentences.txt**
   2. *Find all sentences that contain punctuation marks other than periods.*
      1. **Egrep ‘[A-Za-z][,|!|/|’][A-Za-z]’ sentences.txt**
   3. *Find all sentences that contain punctuation marks, including periods, that are not at the end of the sentence.*
      1. **Egrep ‘[A-Za-z][[:punct:]] [A-Za-z]’ sentences.txt**
   4. *Find all sentences that contain at least three words.*
      1. **Egrep ‘([[:alpha:]]+ [[:alpha:]]+ [[:alpha:]])’ sentences.txt**
   5. *Find all sentences that contain exactly four words.*
      1. **Egrep ‘([[:alpha:]]+ [[:alpha:]]+ [[:alpha:]]+ [[:alpha:]])’ sentences.txt**
   6. *Find all sentences that contain words in which the letters (upper or lower case) a, b, c, and d appear in that order somewhere in the words (not necessarily consecutively). This should, for instance, find “a bat can die”.*
      1. **Egrep ‘([[:alpha:]] [a-d])’ sentences.txt**
3. Using names.txt
   1. *Find all entries of people who live in cities whose names are multiple words.*
      1. **egrep ‘,[A-Z][a-z]+ [A-Z][a-z]+, [A-Z]{2}’ names.txt**
   2. *Find all entries of people whose cities (if given) have at least 8 letters in a row (without a space character in between). Make sure you do not just search for sequences of 8 letters as this could also get last names.*
      1. **Egrep ‘,[A-Z][a-z]{7,}, [A-Z]{2}’ names.txt**
   3. *Find all entries of people who have middle initials.*
      1. **Egrep ‘^[A-Z][a-z]+ [A-Z]\. [A-Z][a-z]+’ names.txt**
   4. *Find all entries of people whose zip code has “22” in it. NOTE: this should not include Mike Keneally whose street address includes “22”.*
      1. **Egrep ‘[0-9]\*22[0-9]\*$’ names.txt**
   5. *Find all entries of people who have complete addresses (including street number and name, city, state and zip code).*
      1. **Egrep ‘^[A-Za-z .]+, [0-9]+ [A-Za-z0-9 #]+, [A-Za-z ]+, [A-Z]{2} \*[0-9]{5}$’ names.txt**
4. Change directory to /usr/share/dict. We will use the Linux dictionary, the file words, for this step.
   1. *Some of the entries consist of words that contain a letter followed by a digit or a digit followed by a letter. Find these and count them (using –c).*
      1. **Egrep ‘^[A-Z0-9]+$’ -c words**
   2. *Find words with that have (at least) 5 consecutive vowels in them (upper or lower case).* 
      1. **Egrep ‘[a|e|i|o|u]{5}|[A|E|I|O|U]{5}’ words**
   3. *Find words that are exactly 12 characters in length (words longer than 12 should not appear).*
      1. **Egrep ‘^[a-zA-Z]{12}$’ words**
   4. *Find all words that have two consecutive punctuation marks.*
      1. **Egrep ‘[[:punct:]]{2}’ words**
   5. *Find all words that have two punctuation marks that are not consecutive.*
      1. **Egrep ‘[[:punct:]][^[:punct:]]+[[:punct:]]’ words**

Shut down your VM if desired, disconnect from the VPN if you are using it, and submit your lab report.