Seth Cram

11/03/2021

CS270: HW3

**Program Log**

* Problem 1:
  + possible strategies:
    - 'cat -n' for add line numbers
    - 'awk' for add and remove line numbers
      * NF is the current line/record number
      * print uses a new line character at the end
      * printf doesn't use a new line character the the end
      * Cant use echo within awk
        + Had to use printf( “\n” )
        + Could’ve probably used just print, but this way was more explicit
    - 'cut' for remove line numbers
      * Use numbers of the word to print as the field
      * Can use number- to say print all numbers after and including this one
  + can't make a file an input and an output in one cmd
  + $Variable works differently in a string with cmd line args
  + Can use -f to check if file, -d to check if directory, and -e to check if exists
  + Verified all above strategies were possible by implementing them
  + If a script errors during file I/O it could overwrite the contents with nothing
* Problem 2:
  + Can give ls a directory to display all the files from
  + First, I tried storing all files located in the directory
    - Wanted to just store filenames, not whole path
      * Use ‘baseline’ cmd to get just filename
    - Stored elements in an array
      * Loop through array with ‘for elementName in “${arrayName[\*]}”
      * Similar to ‘for each’ loop
      * Get val of single elly w/ ${} not just $
    - Used while loop with cnting var instead of for loop
      * Made more sense to me
    - Then look at which had underscores in their names
      * Used ‘tr’ to translate a string into a string w/o any underscores
        + tr options set1 set2
        + Use -d to delete
      * Pattern (set2) must be in single quotes, not double quotes
    - When comparing two strings
      * Have to encase in [[ ]] instead of just [ ]
      * Must enclose operands in “ “
        + Can use wildcards or other reg expression stuff on outside of double quotes
      * Can separate by normal logical operators
  + Grave accents can delay the evaluation of a command
    - Useful for storing in a variable
  + Confused on how to use ‘for..in..do..done’ loops
  + ‘stat’ command used to determine information assoc’d w/ a file
* Problem 3
  + Pass dir or file that you want copied
    - Check whether file or dir
    - mkdir /home/.backup to create backup dir
      * Direct this output /dev/null so user doesnt see
      * Bc throwaway output
  + If file, check if it exists in backup
    - If exists in backup, prompt for user input whether to overwrite or not
      * If not, dont copy in file or file
    - If so, copy in file w/ ‘cp’
  + If dir, check if it exists in backup
    - If exists in backup, prompt for user input whether to overwrite or not
      * If not, dont copy in dir
    - If so, copy in dir w/ ‘cp -r’
      * + Recursively copies in
  + Can use ‘cp -i’ for interactive copy
    - Prompts user w/ overwrite
  + Remove directory and its children w/ ‘rm -r dirName’
    - Recursively removes
    - When used in conjunction with -i, prompts user when overwriting each individual file
      * Tedious
    - Prompt user only once w/ backing up directory
      * Required manual checking for directory and prompting user
      * Worth it
  + Use $HOME to find user’s home directory
    - Could also use $USER to get current user
    - /home/$USER isn’t as simple and definitive as just $HOME
* This assignment took me around 10 hours
  + I was rather inexperienced in shell scripting beforehand
  + Learned a great deal about different commands and their implementations
  + Wasn’t sure to what degree we needed to refine our scripts
    - I did quite a bit, but there’s always room for improvement
  + A large amount of this was spent debugging and documenting
  + Only a small fraction of the time was spent programming
    - I planned in advance for all three scripts
    - Helped implementation move along much quicker
  + Was unsure how to show scripts worked properly
    - Settled on a combination of command line pictures and file contents
    - For the results