

CS 344 – UNIX OS

Fundamentals – Lecture #5

Purushotham Bangalore
Department of Computer and Information
Sciences
University of Alabama at Birmingham (UAB)

uniq

- Uniq discards a duplicate line if adjacent lines in the input are same
- Duplicate lines in the input will not be detected if they are not adjacent
- If input is sorted only one copy of all duplicate and unique lines will be displayed
- Options:
 - To output unique lines only use `-u`
 - To output lines that have duplicates use `-d`
 - To output number of times a line is duplicated use `-c`
 - To ignore first n fields on each input line when comparing use `-f n`

Shell Scripts

- Enables execution of complex tasks by using multiple commands in a single file
- .bashrc or .bash_profile are such examples
- Create simple script using any editor

```
echo "Welcome" $USER
echo "Today's date is: " `date | cut -d' ' -f2-3`
echo "You are logged in to: " `hostname`
echo "There are" `who | wc -l` "user(s) currently logged in"
echo "Your PATH is set to the following directories:"
echo $PATH
```

- To execute a script, it must have execute permission, default file permissions can be get/set using “umask” command

2/21/2005

3

File Permissions

```
$ ls -ld .
drwx--x--x 17 puri  faculty  8192 Feb 20 17:24 .
$ ls -l .bash_rc
-rwx----- 1 puri  staff    1196 Feb 13 19:31 .bash_rc
$ ls -ld /tmp/
drwxrwxrwt 17 root  sys     3218 Feb 20 18:35 /tmp/
$ umask
022
$ touch myfile
$ ls -l myfile
-rw-r--r-- 1 puri  staff    0 Feb 20 18:38 myfile
$ chmod +x myfile
$ ls -l myfile
-rwxr-xr-x 1 puri  staff    0 Feb 20 18:38 myfile
$ umask 027
$ touch newfile
$ ls -l newfile
-rw-r----- 1 puri  staff    0 Feb 20 18:39 newfile
$
```

2/21/2005

4

Executing Shell Scripts

```
$ cat myscript
echo "Welcome" $USER
echo "Today's date is: " `date | cut -d' ' -f2-3`
echo "You are logged in to: " `hostname`
echo "There are" `who | wc -l` "user(s) currently logged in"
echo "Your PATH is set to the following directories:"
echo $PATH
$ ls -l myscript
-rwxr-xr-x 1 puri staff 239 Feb 20 18:23 myscript
$ myscript ← Shell did not find the script in it's PATH
bash: myscript: command not found
$ ./myscript ← You can also specify the full path
Welcome puri
Today's date is: Feb 20
You are logged in to: hestia
There are 1 user(s) currently logged in
Your PATH is set to the following directories:
/usr/bin:/bin:/usr/sbin:/sbin:/hf/local/bin:/mz/mb/puri/bin
$ export PATH=$PATH:.
$ myscript ← You will see the above output this time
```

2/21/2005

5

Creating a complex script

- Read a file – myfile.in
- Output to the screen the total number of unique words
- Output the list of unique words to the file *words.out* along with the number of times each word appears ordered with the most-used words listed first
- Solution:

```
tr -d '?.!:,;' < myfile.in | tr 'A-Z' 'a-z' | tr '\t' '\n\n' \
| sed '/^$/d' \
| sort | uniq -c | sort -rn \
| tee words.out | wc -l
```

2/21/2005

6

Algorithm

- Delete punctuation characters
- Convert all characters to lowercase
- Move each word to a separate line
- Remove blank lines (if any)
- Sort the lines
- Remove duplicates
- Compute word frequency and output to file
- Compute the total number of unique words

2/21/2005

7

Implementation, I

- Delete punctuation characters
 - `tr -d '?!:;,)' < file`
- Convert all characters to lowercase
 - `tr 'A-Z' 'a-z' < file`
- Move each word to a separate line
 - replace space and tab with a newline
 - `tr ' \t' '\n\n' < file`
- Remove blank lines (if any)
 - `sed '/^$/d' file`
 - search for lines starting with `^` and ending `$` with to text in between, and then delete those lines

2/21/2005

8

Implementation, II

- Sort the lines
 - sort file
- Remove duplicates and compute word frequency
 - uniq -c file
- Sort based on word frequency
 - sort -rn file (-n numerical sort, -r reverse sort order)
- Output to file and another utility
 - | tee words.out
- Compute the total number of unique words
 - wc -l

2/21/2005

9

Solution

```
tr -d '?."!:,;' < myfile.in \  
| tr 'A-Z' 'a-z' \  
| tr '\t' '\n\n' \  
| sed '/^$/d' \  
| sort | uniq -c \  
| sort -rn \  
| tee words.out | wc -l
```

← continuation character – no newline

2/21/2005

10