cse15l-lab-reports

Lab 3 - Researching `grep`

Command Format:

```
grep [OPTIONS] PATTERN [FILE...]
```

Note: all sources are cited at the bottom of this document.

Option 1 - `-E` or `--extendedregexp`

This option tells perl to interpret `PATTERN` differently than default. By default, bash interprets the meta characters + ? { | (and) literally, instead of using their meaning in regular expressions, and thus need to be escaped with an escape character to be interpreted properly. This option allows the interpretation of the these characters without having to put an escape character before every single one. Otherwise, the command acts exactly the same as default.

I found this option from the `man` command.

Example 1

```
[cs15lsp23dm@ieng6-203]:biomed:135$ grep -E "B[a-z]{3}[aeiou]+" rr74.txt
          Biotech, Piscataway, NJ, USA). Blots were blocked (0.5%
          CA, USA]; 1:200 anti-iNOS poly [Affinity Bioreagents,
          (Amersham Pharmacia Biotech). INOS-positive control was
          several different antibodies (Affinity Bioreagents and BD
          Transduction Laboratories; Affinity Bioreagents blot
        et al. [ 42]. Because the systemic
```

The regular expression is looking for any sequence that starts with a capital B, followed by any three lowercase alphabetic letters, and ending with any sequence of 1 or more vowels. In this case, my regular expression allowed me to see how many and which lines contained the specific pattern I wanted.

Example 2

```
[cs15lsp23dm@ieng6-203]:biomed:136$ grep -E "[aeiou]{3}" rr74.txt
        circulation, and previous studies using NOS inhibitors [ 1,
        the murine lung following hypoxia, with previous reports [
          (simulating sea level). Exposure was continuous, with
          RVsP was measured as previously described [ 9].
          (100/15 mg/kg), placed supine while spontaneously
          percutaneously into the thorax via a subxyphloid
          radioimmunoprecipitation assay (RIPA) buffer (1×PBS, 1%
          A5441; Sigma-Aldrich, St Louis, MO, USA]). Blots were
        from birth was more severe than we had observed in previous
        than in previous reports, and might have contributed to the
        animals is in agreement with previous reports [ 20,
        peripheral immunolocalization is in accord with previous
        Previously, LeCras
        of hypoxia and inflammation has previously been reported to
        reported previously may be due to the combination of
        we previously demonstrated [ 10] that nNOS does not appear
        species and tissue specific. Previous studies of NOS
        previous studies, the plasma NO metabolite content in the
        Although we attempted to keep the mice continuously
        chain reaction; RIPA = radioimmunoprecipitation assay; RVsP
```

This command matched all lines that had words with 3 vowels consecutively. In this case, the command is useful if you wanted to calculate the rate that you saw words with three consecutive vowels in a random article.

Option 2 - `-c` or `--count`

I found this option using the `man` command.

Example 1

```
[cs15lsp23dm@ieng6-203]:biomed:142$ grep -E -c "[aeiou]{3}[a-z]" rr74.txt
20
```

Instead of printing out all of the lines that matched the regex, grep instead just printed the number of lines it matched. This is useful to filter the output so it can be more readable while also giving you information about the file.

Example 2

```
[cs15lsp23dm@ieng6-203]:biomed:148$ grep -Ec "\s[Zz]" rr74.txt
1
```

This command found all of the lines with a whitespace followed by a capital or lowercase z. This is useful for counting the number of occurrences of words that start with z in a file.

Option 3 - `-n` or `--line-number`

I found this option using the `man` command.

Example 1

```
[cs15lsp23dm@ieng6-203]:biomed:153$ grep -En "ox(ygen|ide)" rr74.txt
146:          Hematocrit and nitric oxide metabolite
223:          Nitric oxide metabolites
387:        hypoxic, a relatively brief period of reoxygenation might
418:        eNOS = endothelial nitric oxide synthase; iNOS =
419:        inducible nitric oxide synthase; nNOS = neuronal nitric
420:        oxide synthase; NO = nitric oxide; NOS = nitric oxide
```

This command finds all matches were either the word oxygen or oxide are used. This is useful to find occurrences of specific words in a file and also which line they occurred on, so you could look at it in context of the file.

Example 2

```
[cs15lsp23dm@ieng6-203]:biomed:154$ grep -En "pressure" rr74.txt
64:          Right ventricular pressure measurements
176:          pressure. As shown in Fig. 1, RVsP was elevated in
248:        increase in pulmonary pressure due to increased viscosity [
333:        ventricular pressure, however, suggesting that iNOS,
423:        = right ventricular systolic pressure.
```

This command finds all occurences of the word “pressure” in rr74.txt, prints the contents of the line containing the match, along with the line number. This acts like a search function in Microsoft Word, allowing you to find and locate occurences of certain words.

Option 4 - `-r` or `--recursive`

I found this option using the `man` command.

Example 1

```
[cs15lsp23dm@ieng6-203]:technical:165$ grep -Erc "\s[ZzXxYy]+" 911report/
911report/chapter-1.txt:85
911report/chapter-10.txt:14
911report/chapter-11.txt:24
911report/chapter-12.txt:56
911report/chapter-13.1.txt:13
911report/chapter-13.2.txt:56
911report/chapter-13.3.txt:66
911report/chapter-13.4.txt:131
911report/chapter-13.5.txt:110
911report/chapter-2.txt:42
911report/chapter-3.txt:114
911report/chapter-5.txt:105
911report/chapter-6.txt:84
911report/chapter-7.txt:63
911report/chapter-8.txt:56
911report/chapter-9.txt:66
911report/preface.txt:2
```

This command looks in all files within the 911report directory for occurrences of a whitespace followed by capital or lowercase z, x, y and prints the number of lines with matches for the specific file. The recursive option is useful in this case for running grep on a lot of files at the same time.

Example 2

```
[cs15lsp23dm@ieng6-203]:technical:179$ grep -Erc "\s[qQ]+" 911report/ | awk -F':' '{sum+=$2;}
1146
```

Going a little bit further with this command, we use grep to find all occurences of a whitespace followed by a capital or lowercase q, and then pipe that into awk which sums up the number for each file that grep outputs. This gives us the sum of all of the lines in all of the files in the directory 911report that presumably have a word that starts with Q. This is useful for getting more information about large textfiles.

Note: I looked in source 2 to figure out how to use awk in this situation.

Sources:

Source 1

Source 2

```
man awk

man grep
```