CS2211b

Software Tools and Systems Programming

Western

UNIVERSITY · CANADA

Week 3a

Pipes

Announcements

Week 2 Q&A on OWL

Lab 2 on OWL

Minor Update to Assignment #1

Quiz Next Lecture!

Midterm Scheduled For:

Date: Saturday March 3rd, 2018

Time: 9:30AM to 11:30AM

Room: WSC-55

Week 2 Practice Questions

3.6 The command **rmdir bar** fails with the message that the directory is not empty. On running **ls bar**, no files are displayed. Why did the **rmdir** command fail?

Week 2 Practice Questions

3.19 Will the command **cp foo bar** work if (i) *foo* is an ordinary file and *bar* is a directory, (ii) both *foo* and *bar* are directories?

Week 2 Practice Questions

4.10 What do you do to ensure that no one is able to see the names of the files you have?

Week 2 Practice Questions

4.14 If umask shows the value (i) 000, (ii) 002, what implications do they have from a security viewpoint?

Week 2 Practice Questions

6.14 You want to concatenate two files, *foo1* and *foo2*, but also insert some text after *foo1* and *foo2* from the terminal. How will you do this?

Week 2 Practice Questions

6.20 (ii) Use command substitution to print the listing of a group of filenames stored in a file.

In-class Activity From Last Week

In-class Activity

Individual Activity: do not have to hand in!

1. Store the text “Hello World!” in the file *hello.txt* using the `echo` command.

2. Concatenate *file1*, *file2* and *file3* into *file4*. Remember that the `cat` command can take multiple files.

3. What is the difference between:

   `wc file`   and   `wc < file`

4. Redirect both the standard output and standard error of the command `ls myfile` to the file *out*.

More on Wild Cards

More on Wild Cards

Some shell support extra wild card characters:

[!…] matches any character NOT between the square brackets
Example: b[!aei]t matches bzt, b3t, b%t, but not bat
Not supported by C Shell

{pat1, pat2, …} matches any pattern or string in braces
Example: {cat,dog,b?t} matches cat, dog, bat, b5t, b-t
Not supported by Bourne Shell

More on Wild Cards

Example:

List any file starting with a non-number character (e.g. bat.txt, _cat.sh, ZAP)

More on Wild Cards

Example:

List any file starting with a non-number character and ending in .txt (e.g. bat.txt, _cat.txt, ZAP.txt)

```
ls [!0-9]*.txt
```

More on Wild Cards

Example:

List any file starting with a non-number character and ending in .txt (e.g. bat.txt, _cat.txt, ZAP.txt)

```
ls [!0-9]*.txt
```

Will not include files that start with . (single dot)

More on Wild Cards

Example:

List any file starting with a non-number character and ending in .txt (e.g. bat.txt, _cat.txt, ZAP.txt)

```
ls [!0-9]*.txt
```

Will not include files that start with . (single dot)

```
ls .[!0-9]*.txt [!0-9]*.txt
```

```
ls {.,}[!0-9]*.txt
```

More on Wild Cards

A note on capitalization:

The characters in character classes like [a-z] or [A-Z] are based on the local settings of the system. For example:

```
[dservos5@cs2211b testdir]$ ls
0  1  2  3  9  a  A  b  B  c  C  z  Z
[dservos5@cs2211b testdir]$ ls [a-z]
a  A  b  B  c  C  z
[dservos5@cs2211b testdir]$ ls [A-Z]
A  b  B  c  C  z  Z
```

May not do what you want in terms of capitalization

Pipes

Western Science

Pipes

- Like redirection but allows us to redirect output streams from one command to input of another.
- Connects standard out of one program to standard in of another program.
- General form:

```
command1 | command2
```

Output of command1 is sent to input of command2

- Examples:

```
who | sort
cat readme.txt | tr –s '[0-9. ()]' ' ' | wc -w
```

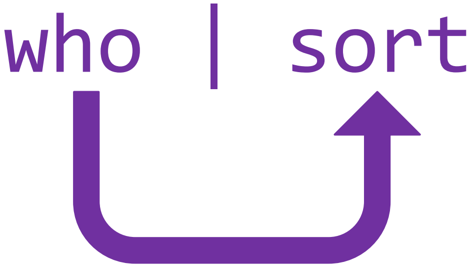
Pipes

Example:

```
who | sort
```

Pipes

Example:



```
who | sort
```

Output of: who

```
yguo379   pts/2        2018-01-22 15:08 (nexus-11.wireless.uwo.ca)
plangary pts/3        2018-01-22 14:50 (nexus-9.wireless.uwo.ca)
dservos5 pts/4        2018-01-22 14:24 (135-23-234-30.cpe.pppoe.ca)
selmolla pts/5        2018-01-22 14:49 (nexus-11.wireless.uwo.ca)
```

- Output of who is sent as input to `sort` command.
- Not displayed on screen.

Pipes

Example:

`who | sort`

Output of: `who | sort`

```
dservos5 pts/4        2018-01-22 14:24 (135-23-234-30.cpe.pppoe.ca)
plangary pts/3        2018-01-22 14:50 (nexus-9.wireless.uwo.ca)
selmolla pts/5        2018-01-22 14:49 (nexus-11.wireless.uwo.ca)
yguo379  pts/2        2018-01-22 15:08 (nexus-11.wireless.uwo.ca)
```

- Output from who command is sorted by the `sort` command.
- The result is displayed via standard output.
- This command sorts the who output by username.

Pipes

Example:

```
cat readme.txt | tr -s '[0-9. ()]' ' ' | wc -w
```

Pipes

Example:

```
cat readme.txt | tr -s '[0-9. ()]' ' ' | wc -w
```

Output of: cat readme.txt

```
Section 6.9 makes use of this feature, while some situations are
presented in Chapter 13 (featuring shell programming).

The system's default then applies (666 for files and 777 for
directories).
```

`cat` reads the file readme.txt and sends it as standard input to the `tr` command.

Pipes

Example:

```
cat readme.txt | tr –s '[0-9. ()]' ' ' | wc -w
```

If a character in SET1 is found multiple times in a row, replace it with just one character from SET2.

Example: If input is "He**ll**o Wor**l**d" and command is: `tr –s l 1`
result is: "He**1**o Wor**1**d" rather than "He**11**o Wor**1**d".

Pipes

Example:

```
cat readme.txt | tr –s '[0-9. ()]' ' ' | wc -w
```

SET1 contains any number from 0 to 9, . (single dot), brackets or a space.

Pipes

Example:

```
cat readme.txt | tr -s '[0-9. ()]' ' ' | wc -w
```

SET2 contains only a space.

Pipes

Example:

```
cat readme.txt | tr –s '[0-9. ()]' ' ' | wc -w
```

Output of: cat readme.txt

```
Section 6.9 makes use of this feature, while some situations are
presented in Chapter 13 (featuring shell programming).

The system's default then applies (666 for files and 777 for
directories).
```

```
tr –s '[0-9. ()]' ' '
```

Pipes

Example:

```
cat readme.txt | tr –s '[0-9. ()]' ' ' | wc -w
```

Output of: cat readme.txt

```
Section 6.9 makes use of this feature, while some situations are
presented in Chapter 13 (featuring shell programming).

The system's default then applies (666 for files and 777 for
directories).
```

```
tr –s '[0-9. ()]' ' '
```

Output of: tr –s '[0-9. ()]' ' '

```
Section makes use of this feature, while some situations are
presented in Chapter featuring shell programming

The system's default then applies for files and for directories
```

Pipes

Example:

```
cat readme.txt | tr -s '[0-9. ()]' ' ' | wc -w
```

Output of: `tr -s '[0-9. ()]' ' '`

```
Section makes use of this feature, while some situations are
presented in Chapter featuring shell programming

The system's default then applies for files and for directories
```

`tr` removes any numbers, periods, and brackets and sends the output to the `wc` command.

Pipes

Example:

```
cat readme.txt | tr -s '[0-9. ()]' ' ' | wc -w
```

↓

26

Pipes

Example:

```
  cat readme.txt | tr -s '[0-9. ()]' ' ' | wc -w
```

26

This command counted the number of words in *readme.txt* excluding numbers, periods or brackets as their own words.

Pipes

With Redirects

- We can combine pipes and redirects:

```
who | sort > sortedlist.txt
cat readme.txt | tr -s '[0-9. ()]' ' ' | wc -w > wordcount.txt
```

  - Stores the output of the example commands in sortedlist.txt or wordcount.txt

- Can also redirect input:

```
tr -s '[0-9. ()]' ' ' < readme.txt | wc -w > wordcount.txt
```

  - `cat` is not needed as the shell reads readme.txt and redirects it to the input stream of `tr`.

Pipes

tee command

- How can we both output to the screen and save the output in a file?
- For example:

```
sort -u nums.txt
```

Displays the unique lines in nums.txt

Pipes

tee command

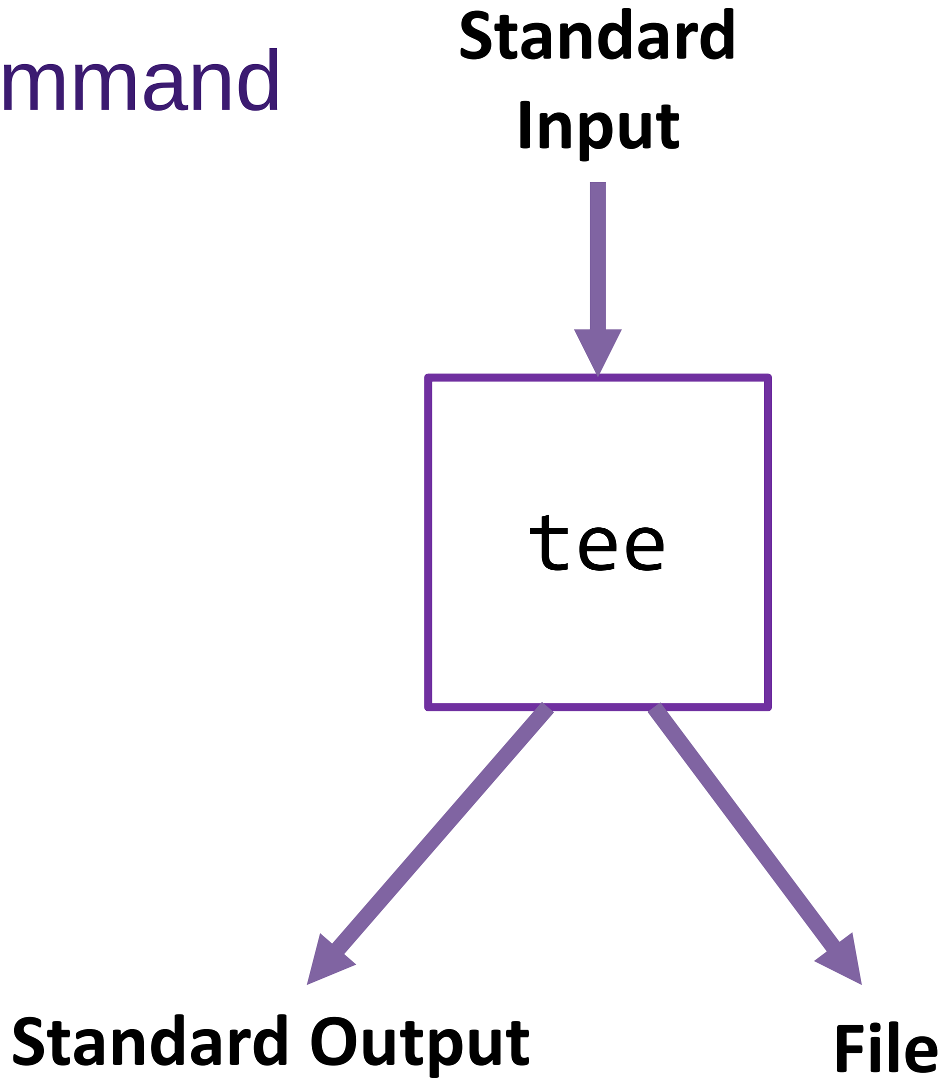
- How can we both output to the screen and save the output in a file?
- For example:

```
sort -u nums.txt | tee output.txt
```

We can use tee command to display and redirect to file.

Pipes

`tee` command



Pipes

tee command

- Can further pipe output of tee.
- For example:

```
sort -u nums.txt | tee output.txt | wc –l
```

Save the unique lines from nums.txt in output.txt and display the number of unique lines to the standard output.

In-class Activity

Using pipes:

1. Write a command to count the number of files in the current directory and both display the result to the screen and save it in out.txt
2. Using the `hostname` command, write a command to replace any number in the `hostname` with the '#' character (e.g. cs2211b would become cs#b) and save the result in out.txt (do not display).
3. Write a command to display the unique lines in *readme.txt*. Remove . (single dots), brackets and numbers. **Hint:** you will need to make every word on it's own line.

/dev/null

- Some times you want to throw away your output and not display it to the screen or save it in a file.
- */dev/null* is a special file that is always empty.
- Any files saved to it or output sent to it is discarded.
- Examples:
  - Copy things to here and they disappear:

    ```
    cp myfile /dev/null
    ```

  - Copy from here and get an empty file:

    ```
    cp /dev/null myfile
    ```

  - Redirect error messages to this file and they are discarded:

    ```
    ls -l fileThatDoesNotExist 2> /dev/null
    ```