Tutorial 03 - Pipelines and Loops

for loop challenge

Challenge: Use the files in `molecules/` to print each # of atoms:

for loop challenge - solution 1

```
for file in *.pdb
do
  cat $file | wc -l
done
```

What could we improve?

for loop challenge - solution 2

```
for file in *.pdb
do
  echo $file
  cat $file | wc -l
done
```

What could we improve?

for loop challenge - solution 3

```
for file in *.pdb
do
  echo $file
  a=$(cat $file | wc -l)
  b=$(($a - 4))
  echo $b
done
```

What could we improve?

for loop challenge - solution 4

```
for file in *.pdb
do
  echo $file | sed 's/.pdb//'
  echo $((`cat $file | wc -l` - 4))
done
```

What could we improve?

echo vs. cat

When to use echo and when to use cat?

- echo returns text, including evaluating shell variables to stdout
- cat access contents of a file and returns it to stdout

echo vs. cat

Very different behaviors that are each useful in their own way.

`echo file.txt` vs. `cat file.txt`

`echo` also useful for “testing” a for loop before actually using it

```
for file in NENE*[AB].txt
do
  echo "bash goostats $file stats-$file"
done
```

```
## bash goostats NENE*[AB].txt stats-NENE*[AB].txt
```

pipes vs. for loops

pipes help you work within files

and

for loops help you work outside of files and within directories

“I am struggling to remember all the functions and flags”

DON’T TRY TO MEMORIZE ALL OF THIS!

- Use google and `man` to look up a function or flag when needed; your TAs and instructor do this all the time...
- Make a cheatsheet!

“How can I get more practice”

- make sure you are working through and typing out the SWC tutorials
- work through the embedded SWC challenges on your own computer before looking at the answers
- work on the challenges in and out of class
- the weekly graded exercises are an important place to practice too

challenges

1. Compile the maximum value from each of Nene's good (A or B, but not Z) data files in `data-shell/north-pacific-gyre/2012-07-3/` into a single file called `NENE_max.txt`.
2. Compile the median value from each of Nene's good (A or B, but not Z) data files in `data-shell/north-pacific-gyre/2012-07-3/` into a single file called `NENE_median.txt`.

solution 1

```
for file in NENE*[AB].txt
do
  cat $file | sort -n | tail -n 1 >> NENE_max.txt
done
```

solution 2

```
for file in NENE*[AB].txt
do
  cat $file | sort -n | head -n 151 | tail -n 1 >> NENE_n
done
```

solution 2

```
for file in NENE*[AB].txt
do
  nlines=$(cat $file | wc -l)
  halflines=$(($nlines / 2 + 1))
  cat $file | sort -n | head -n $halflines | tail -n 1 >>
done
```

function review

`for`

`$`

`history`

`$()`

`$(())`

up arrow

Ctrl-c

Ctrl-a

Ctrl-e