

PIPES, PROGRAMMERS, AND AUTOMATION

Jeff Klow

Cornell College

08 December 2015

"As a programmer, it is your job to put yourself out of business. What you do today can be automated tomorrow."

-Douglas McIlroy

SOME SIMPLE PROGRAMS

Prints 1 through 10 to standard output

```
>  
> ./PrintSequence  
1 2 3 4 5 6 7 8 9 10  
> █
```

Given some arguments, returns their sum

```
>  
> ./sumArgs 1 2 3 4 5 6 7 8 9 10  
The sum is 55.  
> █
```

Source of PrintSequence

```
#include <stdio.h>

int main( int argc, char** argv ) {
    int i;
    for( i = 1; i <= 10; i++ ) {
        printf("%d ", i);
    }
    printf("\n");
} // main
```

Source of sumArgs

```
#include <stdio.h>

int main( int argc, char* argv[] ) {
    int i;
    int sum = 0;
    for( i = 1; i < argc; ++i ) {
        sum += atoi( argv[i] );
    }
    printf( "The sum is %d.\n", sum );
}
```

LAYING PIPES

Basic piping. Note "xargs"

```
>  
> ./PrintSequence | xargs ./sumArgs  
The sum is 55.  
> █
```

FLUSH INTO FILES

You can dump stdout into files too

```
> ./PrintSequence > Numbers.txt
> cat Numbers.txt
1 2 3 4 5 6 7 8 9 10
>
> ./PrintSequence >> Numbers.txt
> cat Numbers.txt
1 2 3 4 5 6 7 8 9 10
1 2 3 4 5 6 7 8 9 10
>
> ./PrintSequence > Numbers.txt
> cat Numbers.txt
1 2 3 4 5 6 7 8 9 10
>
```

Note the difference between `>` and `>>`

FURTHER ADVENTURES IN PLUMBING

You can chain pipes together

```
> pwd
/home/local/CORNELL-COLLEGE/jklow17/Documents/CSC311
> ls -R | grep png > FoundPNGs.txt
> cat FoundPNGs.txt
Piped.png
Sequence.png
SequenceSource.png
Sum.png
SumSource.png
> █
```

MORE FUN THAN PIPE CLEANER ART

Some ways to find/replace text in the pipeline

```
> sed -e "s/png/lol/g" < FoundPNGs.txt > FoundLOLs.txt > ls -R | grep png | sed -e "s/png/lol/g" > FoundPNGs.txt
> cat FoundLOLs.txt > cat FoundPNGs.txt
Piped.lol Piped.lol
PipeIntoFile.lol PipeIntoFile.lol
Pipeline.lol Pipeline.lol
Sequence.lol Sequence.lol
SequenceSource.lol SequenceSource.lol
Sum.lol Sum.lol
SumSource.lol SumSource.lol
> >
```

So long as it reads from stdin and writes to stdout, virtually any program can go in a pipeline.