

# CS2303 In Class Exercises

March 2, 2020

## Abstract

Today's goal is to get practice with pipelining commands in Linux.

Kush Shah kshah2@wpi.edu

Name and WPI email: \_\_\_\_\_

## 1 Pipelining Commands

We talked about the pipelined nature of machine instructions. Now we are considering pipelining of commands in Linux.

(from <https://www.guru99.com/linux-pipe-grep.html>)

The pipe is a command in Linux that lets you use two or more commands such that output of one command serves as input to the next. In short, the output of each process directly as input to the next one like a pipeline. The symbol '|' denotes a pipe.

Example:

```
cat filename | less
```

Here we see the command cat, the idea filename, which would be replaced, when this example is tried, with the name of a file accessible from the current working directory, we see the literal symbol vertical bar, referred to as a pipe, and the command less.

The idea is that cat operating on the specific file produces output, and that output is to be used as input for the command less.

The command cat by itself will output a file to the terminal. See Figure 1.

Some files are so big, that cat by itself is undesirable. We can provide the output of cat as input to the command less.

Once within the environment provided by the command less, there are possibilities.

You can use the up and down arrow keys to move line by line.

If you want to move page by page, use space key to move to next page and 'b' key to go back to the previous page.

If you want to move to the beginning of the file, use 'g' key. If you want to go to the end of the file, press 'G' key.

To summarize:

```
vagrant@vagrant: ~  
who struggled here,  
have consecrated it,  
far above our poor power to add or detract.  
The world will little note, nor long remember what we say here,  
but it can never forget what they did here.  
It is for us the living, rather,  
to be dedicated here to the unfinished work  
which they who fought here  
have thus far so nobly advanced.  
It is rather for us  
to be here dedicated  
to the great task remaining before us --  
that from these honored dead  
we take increased devotion  
to that cause  
for which they gave the last full measure of devotion --  
that we here  
highly resolve  
that these dead shall not have died in vain --  
that this nation, under God,  
shall have a new birth of freedom --  
and that government  
of the people,  
by the people,  
for the people,  
shall not perish from the earth.  
Abraham Lincoln  
November 19, 1863  
vagrant@vagrant:~$
```

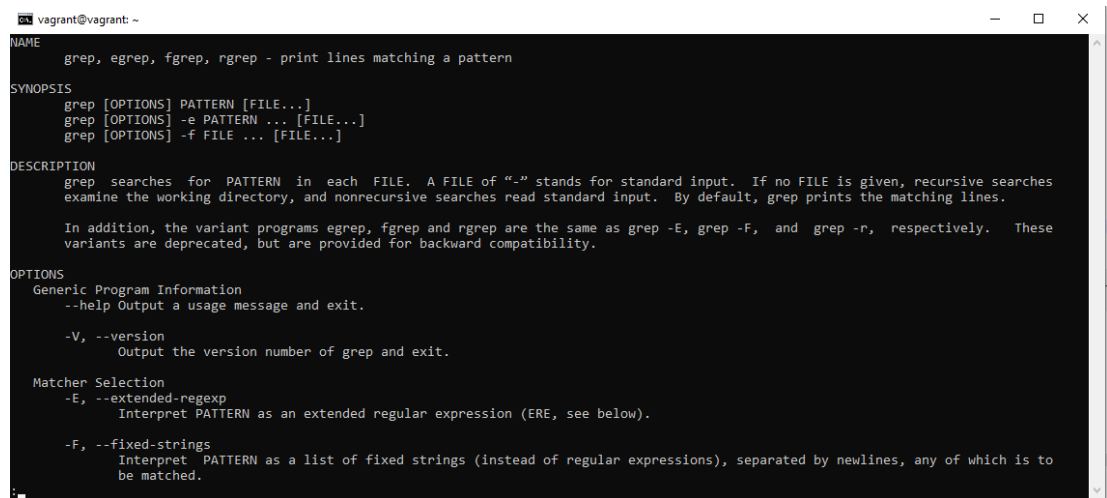
Figure 1: Gettysburg Address, Bliss version

```
❏ vagrant@vagrant: ~  
who struggled here,  
have consecrated it,  
far above our poor power to add or detract.  
The world will little note, nor long remember what we say here,  
but it can never forget what they did here.  
It is for us the living, rather,  
to be dedicated here to the unfinished work  
which they who fought here  
have thus far so nobly advanced.  
It is rather for us  
to be here dedicated  
to the great task remaining before us --  
that from these honored dead  
we take increased devotion  
to that cause  
for which they gave the last full measure of devotion --  
that we here  
highly resolve  
that these dead shall not have died in vain --  
that this nation, under God,  
shall have a new birth of freedom --  
and that government  
of the people,  
by the people,  
for the people,  
shall not perish from the earth.  
Abraham Lincoln  
November 19, 1863  
vagrant@vagrant:~$ cat blissGA | less_
```

Figure 2: Output from cat can be piped to less.

```
vagrant@vagrant: ~  
Four score and seven years ago  
our fathers brought forth on this continent a new nation,  
conceived in liberty, and dedicated to the proposition that  
all men are created equal.  
Now we are engaged in a great civil war,  
testing whether that nation,  
or any nation so conceived and so dedicated,  
can long endure.  
We are met on a great battlefield of that war.  
We have come to dedicate a portion of that field,  
as a final resting place for those  
who here gave their lives  
that that nation might live.  
It is altogether fitting and proper that we should do this.  
  
But, in a larger sense, we can not dedicate --  
we cannot consecrate --  
we can not hallow -- this ground.  
The brave men, living and dead,  
who struggled here,  
have consecrated it,  
far above our poor power to add or detract.  
The world will little note, nor long remember what we say here,  
but it can never forget what they did here.  
It is for us the living, rather,  
to be dedicated here to the unfinished work  
which they who fought here  
have thus far so nobly advanced.  
:
```

Figure 3: Notice that the beginning of the file is displayed, and that there is a colon, prompt to the command less, at the bottom of the terminal window.



```
vagrant@vagrant: ~
NAME
    grep, egrep, fgrep, rgrep - print lines matching a pattern

SYNOPSIS
    grep [OPTIONS] PATTERN [FILE...]
    grep [OPTIONS] -e PATTERN ... [FILE...]
    grep [OPTIONS] -f FILE ... [FILE...]

DESCRIPTION
    grep searches for PATTERN in each FILE. A FILE of "-" stands for standard input. If no FILE is given, recursive searches
    examine the working directory, and nonrecursive searches read standard input. By default, grep prints the matching lines.

    In addition, the variant programs egrep, fgrep and rgrep are the same as grep -E, grep -F, and grep -r, respectively. These
    variants are deprecated, but are provided for backward compatibility.

OPTIONS
    Generic Program Information
    --help Output a usage message and exit.

    -V, --version
        Output the version number of grep and exit.

    Matcher Selection
    -E, --extended-regexp
        Interpret PATTERN as an extended regular expression (ERE, see below).

    -F, --fixed-strings
        Interpret PATTERN as a list of fixed strings (instead of regular expressions), separated by newlines, any of which is to
        be matched.
```

Figure 4: Man page output can be provided as input to less.

Up arrow – Move one line up  
Down arrow – Move one line down  
Space or PgDn – Move one page down  
b or PgUp – Move one page up  
g – Move to the beginning of the file  
G – Move to the end of the file  
ng – Move to the nth line

It is possible to read man pages with less. See Figure 4.

Exercises: (As always, if you have questions, ask. After this occasion, this material becomes background knowledge.)

1. Describe how you would obtain information from the man pages on the function grep.

`man grep | less`

```
vagrant@vagrant:~$ cat blissGA | grep -i we | less_
```

Figure 5: We can create a three stage pipeline.

```
vagrant@vagrant: ~  
Now we are engaged in a great civil war,Now we are engaged in a great civil war,  
We are met on a great battlefield of that war.  
We have come to dedicate a portion of that field,  
It is altogether fitting and proper that we should do this.  
But, in a larger sense, we can not dedicate --  
we cannot consecrate --  
we can not hallow -- this ground.  
far above our poor power to add or detract.  
The world will little note, nor long remember what we say here,  
we take increased devotion  
that we here  
(END)
```

Figure 6: The filter text “we” is found within the word “power”.

It is possible for multiple stages to exist in a pipeline.

We can use the command `grep` to filter. We can `cat` a file piping it to `grep`, and tell `grep` what to filter from that file.

We can use the command `wc` to count things. “`wc -l`” counts the number of lines.

Note that creation of these pipes is a kind of programming.

<https://www.shellscript.sh/tips/pipelines/>

Exercises: (As always, if you have questions, ask. After this occasion, this material becomes background knowledge.)

1. Create a three stage pipeline.

```
cat MyFile.txt | grep -i hello | less
```

```

vagrant@vagrant:~$ cat blissGA | grep -i 'we '
Now we are engaged in a great civil war,
We are met on a great battlefield of that war.
We have come to dedicate a portion of that field,
It is altogether fitting and proper that we should do this.
But, in a larger sense, we can not dedicate --
we cannot consecrate --
we can not hallow -- this ground.
The world will little note, nor long remember what we say here,
we take increased devotion
that we here
vagrant@vagrant:~$ cat blissGA | grep -i 'we ' | wc -l
10
vagrant@vagrant:~$

```

Figure 7: We can filter and then count.

```

vagrant@vagrant:~$ vi hello-world.sh
vagrant@vagrant:~$ vagrant@vagrant:~$ ls
blissGA hello-world.sh makeDemoText
vagrant@vagrant:~$ ls -l
total 12
-rw-rw-r-- 1 vagrant vagrant 1532 Mar  1 21:03 blissGA
-rw-rw-r-- 1 vagrant vagrant   33 Mar  2 13:58 hello-world.sh
-rw-rw-r-- 1 vagrant vagrant   87 Mar  1 20:32 makeDemoText
vagrant@vagrant:~$

```

Figure 8: The editor used here, vi, pops up its own window while in use.

## 1.1 Shell Scripts

Another way of programming in Linux is writing shell scripts. Shell scripts can be prepared using an editor.

(from <https://www.ubuntupit.com/simple-yet-effective-linux-shell-script-examples/>)

An example shell script is:

```

#!/bin/bash
echo "Hello World"

```

This script declares itself to be for use in the shell named bash (commonly available). It uses the single command “echo”, which supports a parameter that can be a literal string.

When you save and quit from the editor, the file will have been created. The permissions of the file will not include being able to execute.

To change the permissions to permit execution, type

```
$ chmod a+x hello-world.sh
```

or

```

-rw-rw-r-- 1 vagrant vagrant 87 Mar 1 20:32 makeDemoText
vagrant@vagrant:~$ chmod 555 hello-world.sh
vagrant@vagrant:~$ ls -l
total 12
-rw-rw-r-- 1 vagrant vagrant 1532 Mar 1 21:03 blissGA
-r-xr-xr-x 1 vagrant vagrant 33 Mar 2 13:58 hello-world.sh
-rw-rw-r-- 1 vagrant vagrant 87 Mar 1 20:32 makeDemoText
vagrant@vagrant:~$

```

Figure 9: Once the file has been made executable, we can invoke it.

```

vagrant@vagrant:~$ ./hello-world.sh
hello world
vagrant@vagrant:~$

```

Figure 10: We can execute it.

```
chmod 555 hello-world.sh
```

Exercises: (As always, if you have questions, ask. After this occasion, this material becomes background knowledge.)

1. Change the same shell script to produce some other output.

```
echo "Goodbye"
```

2. Write a shell script that writes "These are my files", and then lists only those files in your current directory that contain the letter grouping "we".

```
echo "These are my files"
ls | grep -i we
```



We can write scripts containing while loops.

```
#!/bin/bash
i=0

while [ $i -le 2 ]
do
echo Number: $i
((i++))
done
```

Notice that to get the value of the variable i, we write “\$i”.

We can write scripts containing for loops.

```
#!/bin/bash

for (( counter=1; counter<=10; counter++ ))
do
echo -n "$counter "
done

printf "\n"
```

## 2 Questions about this, or about the practice final