Laboratory Experiment III

Input and Output Control

by Alex Shaw III

Unedited Version

April 5, 2005

TABLE OF CONTENTS

INTRODUCTION

This experiment involves working with redirection tools, pipes, tees, and standard input and output. It focuses on shell operations, input redirection, and output redirection.

1
LABORATORY SHEETS

LABORATORY EXPERIMENT III

Procedures

LEHMAN COLLEGE
of the City University of New York

Experiment 3 Redirection, Pipes, Tees, and Standard I/O

I. Objectives: To study the Linux command flow

II. Procedure:

A. Basic Shell Operation

1. Login to the Linux system on your own working directory.
2. Issue the following commands:
   - who
   - ls –l
   - date

   How were the commands issued? What was used to enter the commands? Where did the output appear on each of the commands?

B. Output Redirection

1. Issue the following commands:
   - who >users
   - ls –l >list
   - date >now
2. Issue the command ls –l.
   Do you see the files users, list, and day?
3. Issue the "cat" command on each of the aforementioned files. Describe what you saw in each of the files. Explain why the output did not appear on the screen. Explain the delimiter ">".
4. Issue the command:

   echo I am happy
5. Issue the command:

   echo "I am happy" >xyz

↪ continue on next page…

Did the command appear on the screen at any time? Did the results appear both times? Explain in detail. Compare the echo command with the previous commands.

6. Create a file "testcase" using the vi editor.
   - Enter data into the "testcase" file (more than one line).
   - Save the file "testcase".

7. Issue the command "cat testcase".
8. Issue the command "cat testcase >mycase". Explain the difference between the two.

C. Input Redirection

1. Issue the command:
   wc –l users

2. Issue the command:
   wc –l <users

3. Issue the command:
   wc –l testcase

4. Issue the command:
   wc –l <mycase

5. Analyze the results of the aforementioned and explain, in detail, the results of the commands. Explain and discuss the delimiter "<".

D. Pipes

1. Issue the command:
   who | wc –l

   What can you say about the output? Discuss the "|" delimiter. Explain the difference between the pipe and redirection delimiters.

2. Issue the command:
   who | sort >orderusers

3. Issue the command:
   who |sort >orderusers |wc –l

4. Issue the command:

   who |sort >orderusers |wc –l >analyze

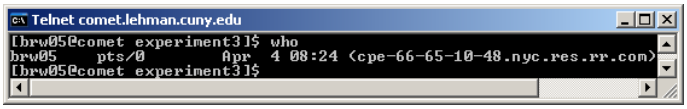
5. Explain the results, in detail, of the previous commands (NOTE: A new command was introduced).

E. Tees

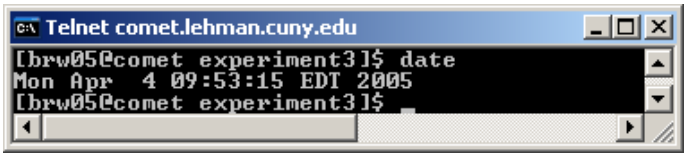
1. Issue the commands:
   who |grep ba(or bb) |tee classmates |wc –l >matenumber.
2. Explain, in detail, the output matenumber. Explain, in detail, the file, classmates.

2
DATA SHEETS

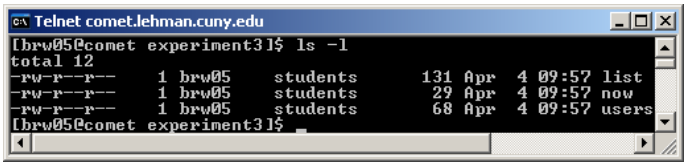
LABORATORY EXPERIMENT III



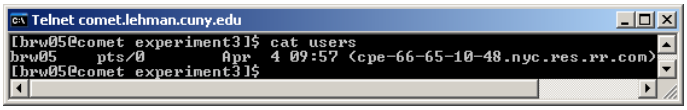
Output 2-1 Results of the **who** command, displaying one user currently logged-in to the system.



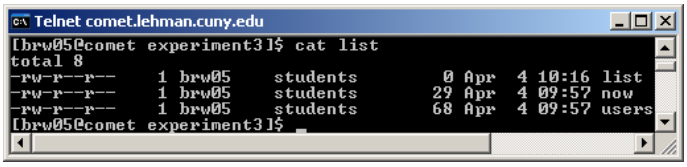
Output 2-2 Results of the **date** command.



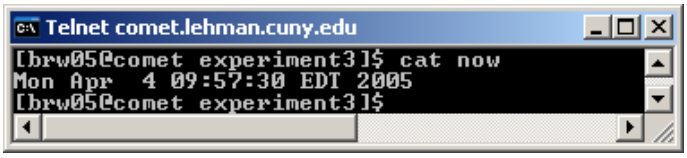
Output 2-3 Results of the **ls -l** command.



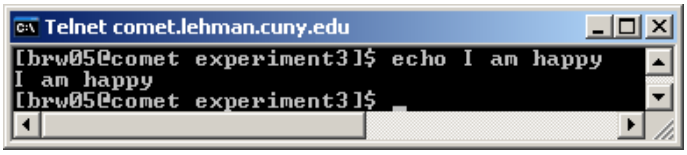
Output 2-4 Results of the **cat *users*** command.



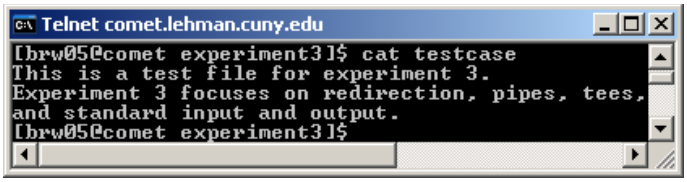
Output 2-5 Results of the **cat *list*** command.



Output 2-6 Results of the **cat *now*** command.



Output 2-7 Results of the **echo *I am happy*** command.



Output 2-8 Results of the **cat *testcase*** command after creating the *testcase* file with **vi**.

Results

Below are results of the experiment, section-by-section; additional remarks are in the Analysis section.

A test directory (experiment3) was created to perform the procedures in this experiment.

Basic Shell Operation

Using telnet and keyboard input, **who** (**Output 2-1**), **ls -l**, and **date** (**Output 2-2**) produced its output to the screen.

Output Redirection

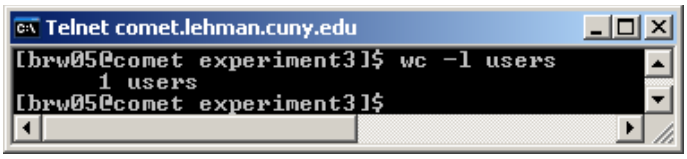
After issuing the **who** > ***users***, **ls -l** > ***list***, and **date** > ***now*** commands, the **ls -l** (**Output 2-3**) command does not display the files—*users*, *list*, and *day*. Instead, it displays the files—*users*, *list*, and *now*, in ascending order.[1]

Results of **cat *users***, **cat *list***, and **cat *now*** are show in **Output 2-4**, **Output 2-5**, and **Output 2-6**.[1]

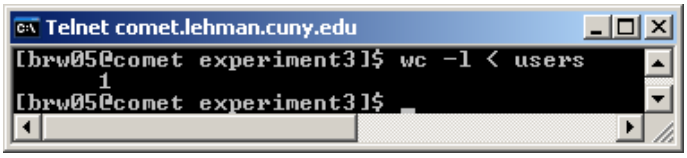
The **echo *I am happy*** (**Output 2-7**) command issues output directly to the monitor. On the other hand, **echo "*I am happy*" > *xyz*** does not appear on the monitor.[1]

The difference between **cat *testcase*** (**Output 2-8**) and **cat *testcase* > *mycase*** are discussed in the Analysis section.

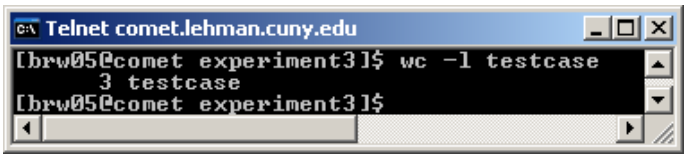
[1] See the Analysis section for an in-depth explanation and additional information.



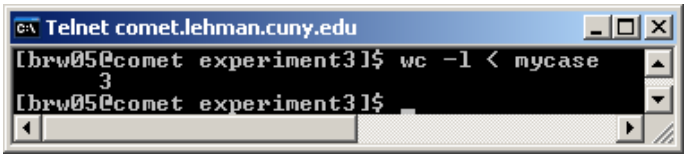
Output 2-9 Results of the **wc -l *users*** command.



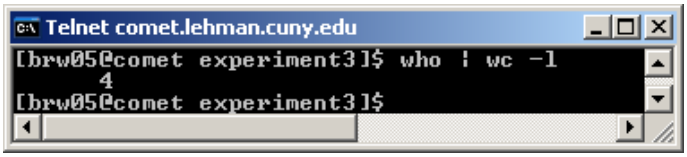
Output 2-10 Results of the **wc -l < *users*** command.



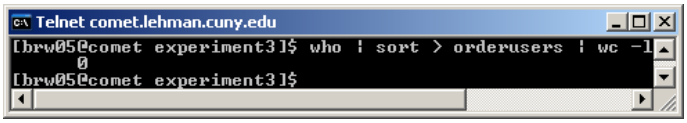
Output 2-11 Results of the **wc -l *testcase*** command.



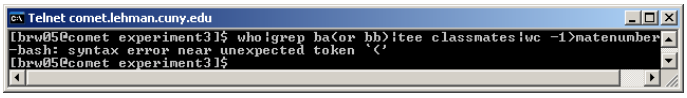
Output 2-12 Results of the **wc -l < *mycase*** command.



Output 2-13 Results of the **who | wc -l** command.



Output 2-14 Results of the **who | sort > *orderusers* | wc -l** command.



Output 2-15 Results of the
who|grep ba(or bb)|tee *classmates*|wc -l>*matenumber* command.

Input Redirection[1]

The **wc -l *users*** (**Output 2-9**) command displays the number of lines in the *users* file.

The **wc -l < *users*** (**Output 2-10**) command sends the number of lines in the *users* file to the **wc -l** command, which in turn displays 'only' the line count.

The **wc -l *testcase*** (**Output 2-11**) command displays the number of lines in the *testcase* file.

The **wc -l < my*case*[2]** (**Output 2-12**)command sends the number of lines in the *mycase* file to the **wc -l** command, which in turn displays 'only' the line count.

Pipes[1]

Results of **who | wc -l** are shown in **Output** 2-13. It routes the results of **who** to **wc -l**, which in turn displays the number of lines generated by **who**.

The **who | sort > *orderusers* | wc -l**[1] displays results shown in **Output 2-14**.

Tees

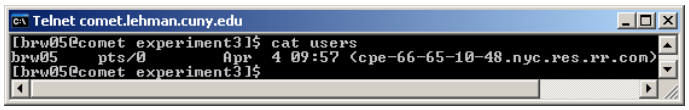
The command
who|grep ba(or bb)|tee *classmates*|wc -l>*matenumber* generates an error (**Output 2-15**)[1].

[1] See the Analysis section for more details.

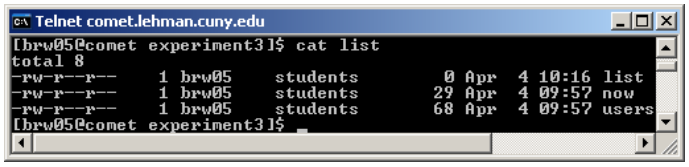
[2] The *mycase* file was created earlier with the **cat *testcase* > *mycase*** command.

3
ANALYSIS

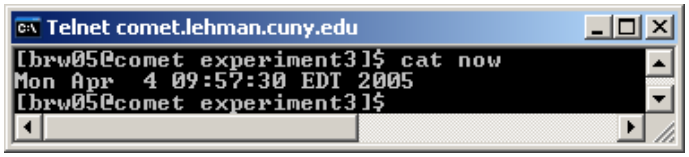
LABORATORY EXPERIMENT III



Output 3-1 Results of the **cat *users*** command.



Output 3-2 Results of the **cat *list*** command.



Output 3-3 Results of the **cat *now*** command.

Explanations

This section discusses any discoveries, including new commands, and solutions used to resolve any errors encountered during the experiment.

The steps below refer to the step of the particular section found on the lab sheets.

Basic Shell Operation

Discovery—Step 1

On the lab sheet, it appears as if **who>*users***, **ls-l>*list***, and **date>*now*** contain no spaces. After performing these commands, with and without spaces, it is determined that spaces are required between commands and options. However, spaces are not required between commands and delimiters, or delimiters and options. Therefore, **who>*users*** and **date>*now*** function using both methods; **ls -l>*list***, on the other hand, requires a space between the **ls** command and the **-l** option.

Output Redirection

Error—Step 2

After issuing **who>*users***, **ls -l>*list***, and **date>*now*** in Step 1, the question "Do you see the files *users*, *list*, and *day*?" after issuing the **ls -l** command was asked. The *day* file did not appear in the list. Perhaps *day* could had been mistaken for *now*.

Explanation—Step 3

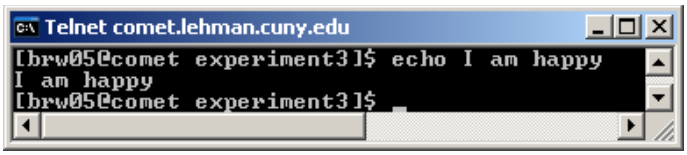
The *users* file contains users who were logged in at the time the command **who>users** was issued (**Output** 3-1).

The *list* file contains information on the directory contents obtained from the **ls -l** command (**Output** 3-2).

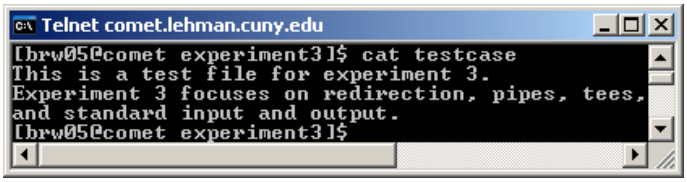
The *now* file contains the day, month, time, region, and year, obtained from the **date>*now*** command (**Output** 3-3).

Output from the **cat** command performed on the *users*, *list*, and *now* files did not send information to the monitor because the > delimiter was used.

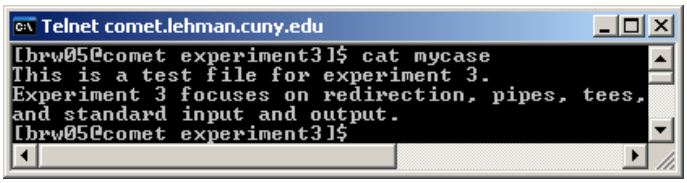
The > delimiter functions as an output redirection tool. It transfers the results of a command to a file.



Output 3-4 Results of the **echo *I am happy*** command.



Output 3-5 Results of the **cat *testcase*** command after creating the *testcase* file with **vi**.



Output 3-6 Results of the **cat *mycase*** command after issuing a **cat *testcase* > *mycase*** command.

Explanation—Step 5

The **echo *I am happy*** command sends output directly to the monitor (**Output 3-4**).

The **echo "*I am happy*">*xyz*** command sends output to the *xyz* file.

Below is a comparison between **echo** and **cat** as it pertains to the experiment:

echo	**cat**
Sends information to the monitor directly from keyboard input	Displays the contents of a file to the screen
Can redirect output to a file	Can redirect output to a file
Cannot accept input redirection	Can accept input redirection

Explanation—Step 8

The **cat *testcase*** command displays the contents of *testcase* (**Output 3-5**).

The **cat *testcase*>*mycase*** command redirects output from **cat *testcase*** to the *mycase* (**Output 3-6**) file. It overwrites *mycase* if it exists; otherwise, it creates *mycase*. Thus, *mycase* contains the same information as *testcase*.

The main difference between the two is that the first command outputs to the monitor and the second to a file.

Input Redirection

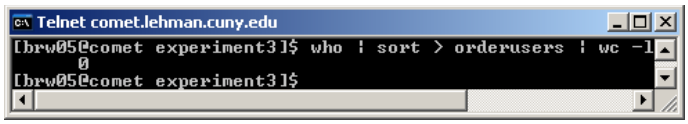
Explanation—Step 5

The table below represents examples of input redirection:

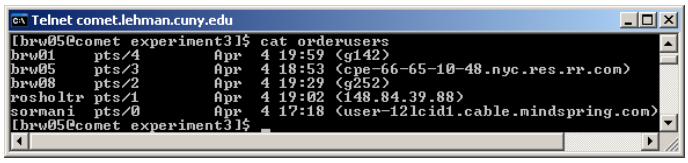
Command[1]	**Explanation**
wc -l *users*	Displays the line count of the *users* file
wc -l < *users*	Executes **wc -l** based on input redirection from the *users* file, not the keyboard
wc -l *testcase*	Displays the line count of the *testcase* file
wc -l < *mycase*	Executes **wc -l** based on input redirection from the *mycase* file, not the keyboard

The < delimiter represents input redirection. Instead of using the keyboard for input, the command receives its input from a file.

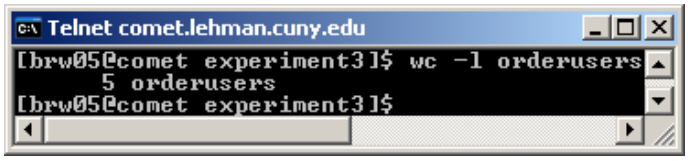
[1] The results of these commands are on page 2.



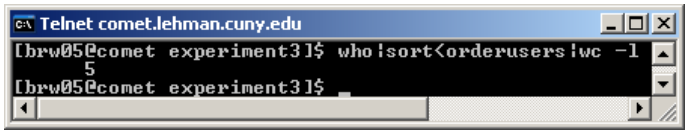
Output 3-7 Results of the **who | sort > *orderusers* | wc -l** command.



Output 3-8 Results of **cat orderusers** after issuing the **who | sort > *orderusers*** command.



Output 3-9 Results of the **wc -l orderusers** command.



Output 3-10 Results of the **who | sort < orderusers | wc -l**command.

Pipes

Explanation—Step 1

In **who | wc -l**, the results of **who** is transferred to **wc -l**, which in turn displays the numbers of lines generated by **who**.

The | delimiter delivers the results of one command to be processed by another command.

Error—Step 3

The command **who | sort > *orderusers* | wc -l** produces 0 as a result (**Output 3-7**). However, this is an error because **who | sort > *orderusers*** returns the results shown in **Output 3-8** and **wc -l *orderusers*** returns results shown in **Output 3-9**. Therefore, both results combined should sort *orderusers* in ascending order and display its line count.

Pipes normally work with commands. The first pipe—**who** to **sort** works fine; however, the last pipe attempts to take the sorted *orderusers* file and pipe it to the **wc -l** command. To fix the problem, *orderusers* must be redirected into **sort**, then **sort** can be piped with **wc -l** because the pipes are generated between two commands.

The correct syntax to achieve the desired result is, **who | sort < *orderusers* | wc -l** (**Output 3-10**).

Error—Step 4

Although an error message does not appear, the intended purpose of the **who | sort > *orderusers* | wc -l>*analyze*** is to transfer the results in **Output 3-10** to the *analyze* file. The same solution is applied as in Step 3.

The correct syntax to achieve the desired result is, **who | sort < *orderusers* | wc -l>analyze**.

Tees

Error—Step 2

The **who|grep ba(or bb)|tee *classmates*|wc -l>*matenumber*** command makes an attempt to search for the string 'ba' or 'bb' in the resulting **who** command. It then pipes the result in the *classmates* file. Finally, the **tee** results are piped to **wc -l**, which in turn stores the line count of *classmates* to *matenumber*.

The correct syntax to achieve the desired result is, **who|grep -E "ba|bb"|tee *classmates*|wc -l>*matenumber*[1]**

[1] Unix Manual Page for grep: http://www.scit.wlv.ac.uk/cgi-bin/mansec?1+grep

New Commands[1]

During the experiment, **sort**, **grep**, and **tee** were introduced. Below is a brief explanation and syntax of each command:

Command	Syntax	Explanation
sort	sort [option]… [file]…	Writes sorted information to standard output
grep	grep [option]… pattern [file]…	Search for pattern in each *file* or standard input
tee	tee [option]… [file]…	Copy standard input to each *file*, and to the standard output

[1] The syntax and explanation for each command is found using the help option in the Bourne Again Shell. Example: **tee --help**

References

1. Sarwar, Syed Mansoor, Robert Koretsky, Syed Aqeel Sarwar. Linux: The Textbook. Boston: Addison Wesley Longman Inc., 2002.