**Assignment 4**

**Deliverables**

* This lab assignment is divided into 5 tasks and one review exercise.
* Use the same document to answer questions and add screenshots.
* After completing all 5 tasks and review exercise show this document to your lab professors for grading.
* Demo your work to your lab professor during lab class for grades.
* Make sure of your work when you visit your professor for a demo.

“Unanswered questions, missing screenshots, or incorrect screenshots will be counted as a mistake.

* (No submission is accepted; the student must demo during lab class).
* Due by the start of next Lab class.

**NOTE:**

* ***Read all the instructions carefully, directory and file names have to be exactly the same as specified in the instructions.***
* **Spelling and capitalization must be Exactly as shown**

**Assignment’s TASKS:**

For Tasks 1,2, 3 and review exercise (Total 15 points, **you lose 1 point for each mistake**)

\*\*Wrong file or directory name will be considered a mistake\*\*.

**Task 1**

Recall Output Redirection using the > character, where you can save the output of a Linux command line in a file instead of having the output displayed on your screen.

1. Try each of the Linux command lines written below at the bash shell prompt. Do not type the leading dollar ($) prompt character. Only type the command name and any arguments to the command, separated by spaces.

**$ date**

**$ users**

**$ who**

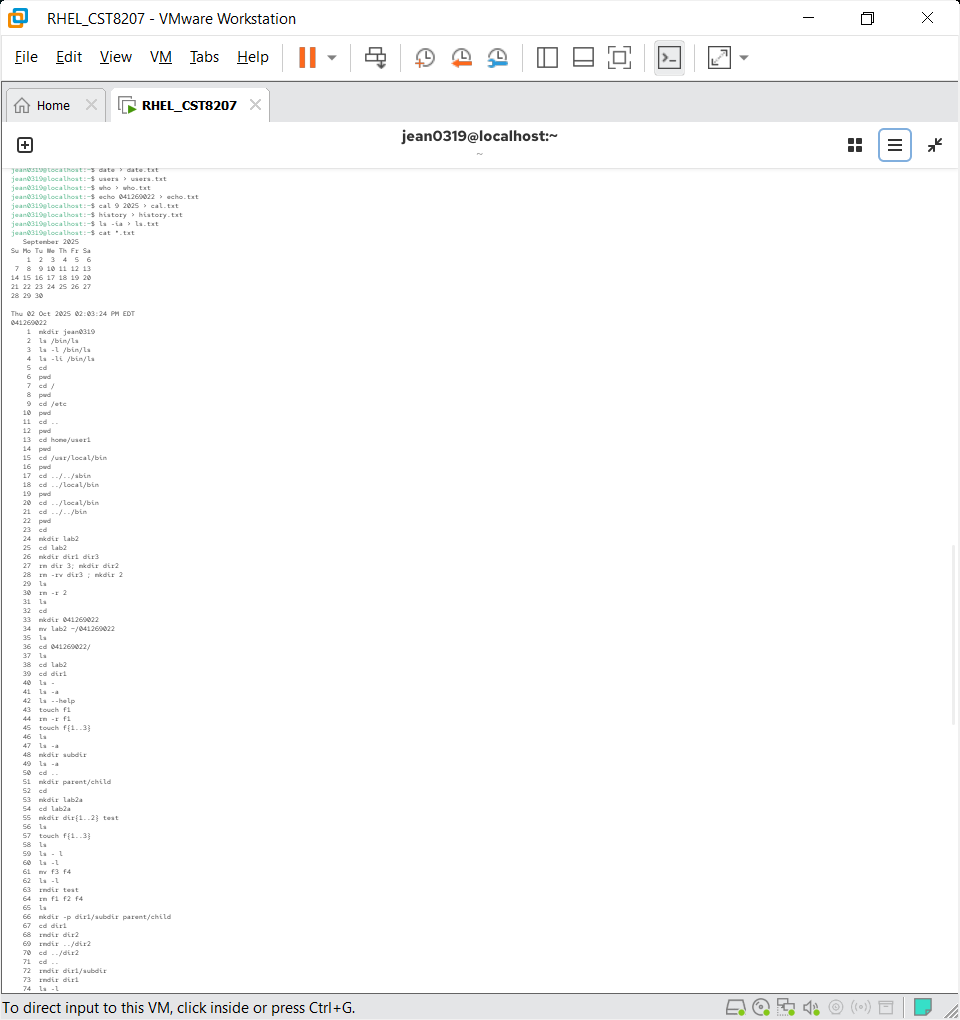
**$ echo your\_student\_number**

**$ cal 9 2024**

**$ history**

**$ ls -ia**

**Screenshot**

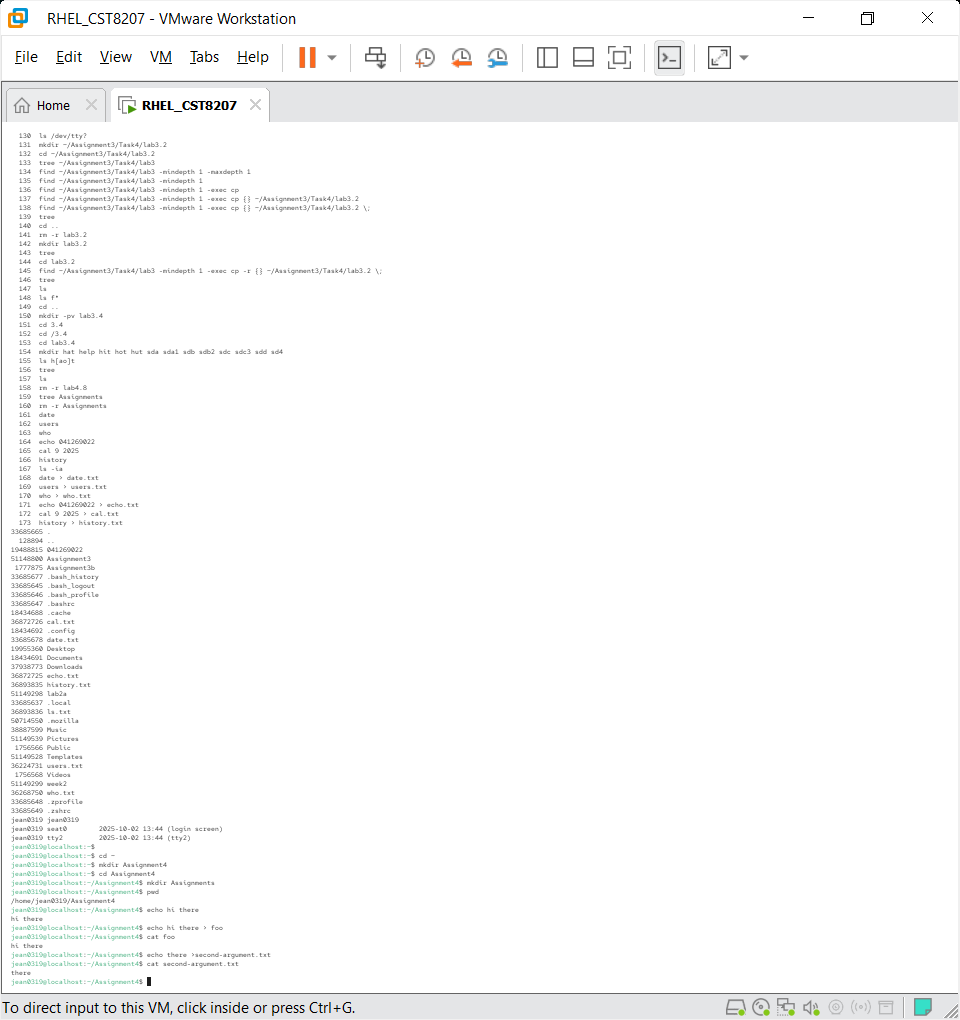
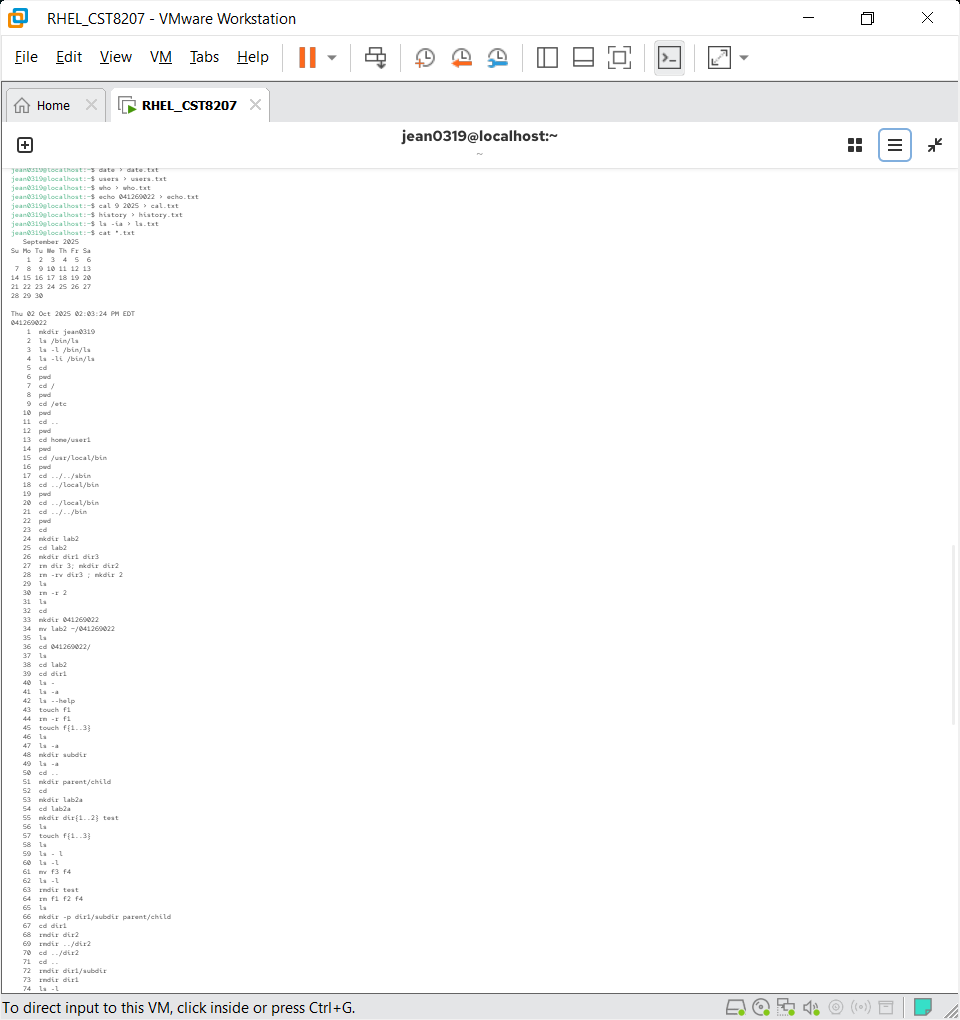


1. Now, execute all of the above Linux commands with the same command arguments, and **redirect the output of each of the commands into a file that is the name of the command with a .txt extension added**

**Example for file name: commandname.txt**

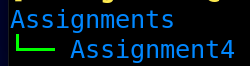
View all the files you created above in one command line so that your output should show the filename corresponding to its content.

**Screenshot of command lines.**

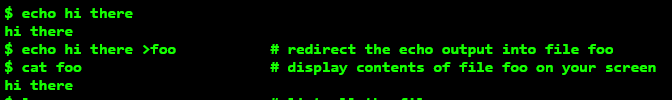
****

**Task 2**

Create the following directory structure in **your HOME** (user personal home directory) directory.



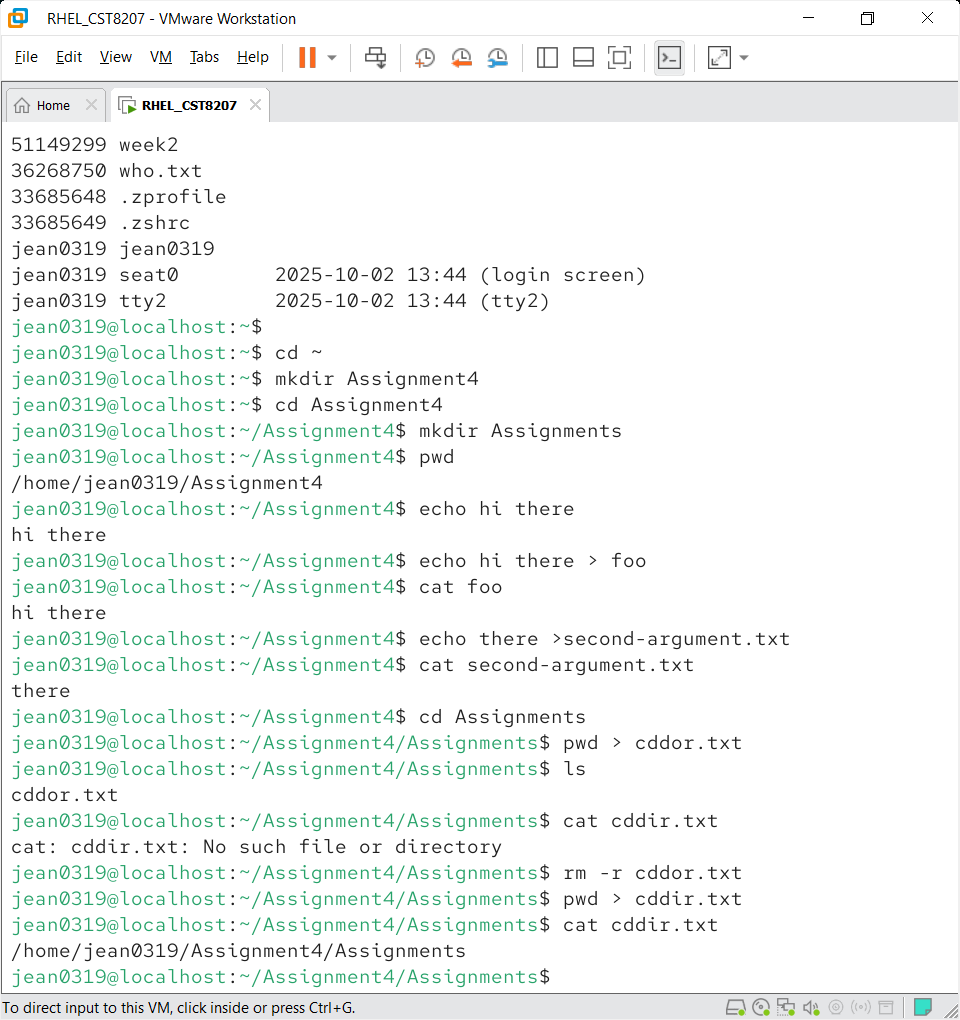
Now make Assignment4 as your current working directory and execute the commands shown in the below screenshot.



1. Looking at the above command line that uses **output redirection**, type a new echo command line that uses only the second argument (not both arguments) from that command line, and redirect the output (just the second argument from above) into a file named **second-argument.txt**. The output file should contain only one word.

**Hint: What is the second argument in that command line?**

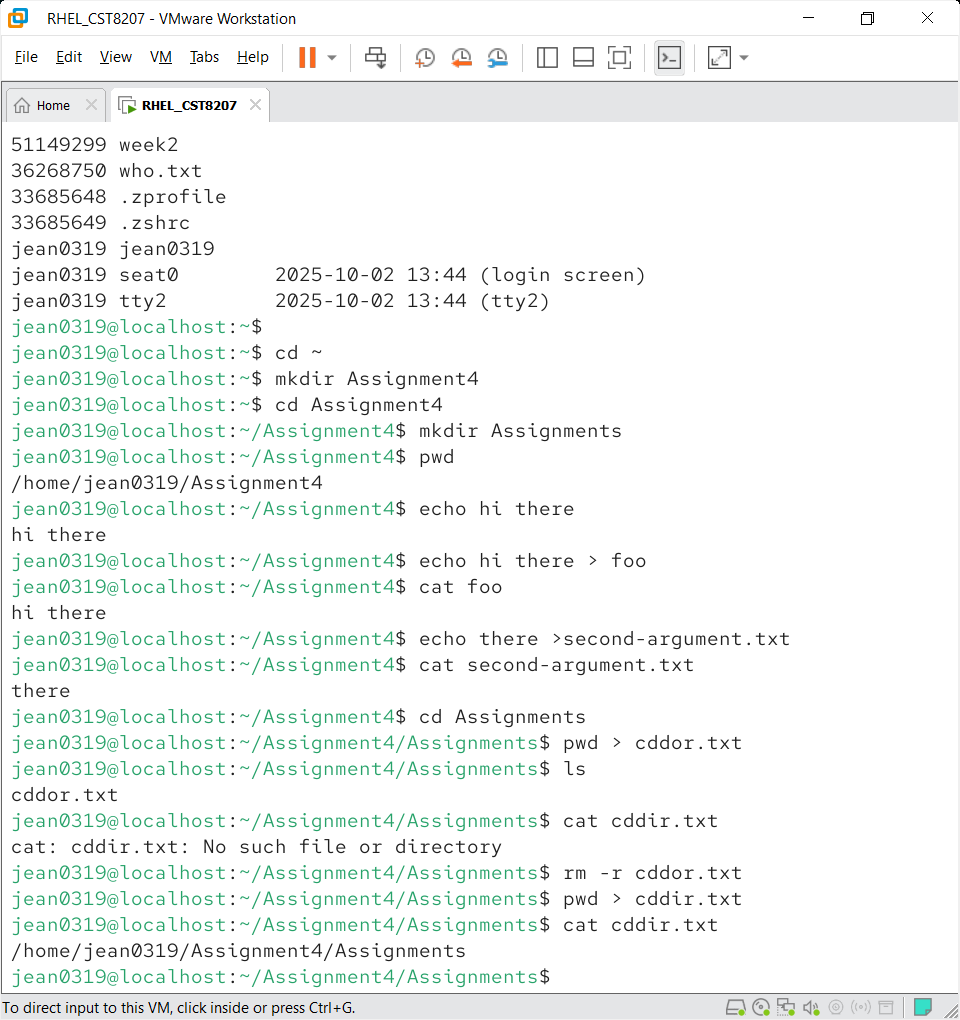
**Add a screenshot of the content of the file second-argument.txt**

****

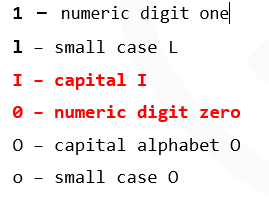
* Now make the Assignments directory your current working directory.

1. Run the command **pwd** and redirect the output of the command into the file **cddir.txt** in the directory Assignments

**Add a screenshot of the content of the file cddir.txt**

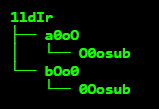
****

**Task 3**

**\*\* Have correct name, exactly same as specified in instructions. Failed to use the correct name leads to loose points.**

1. Create the following directory structure under directory Assignment4. Spelling and capitalization must be exactly as shown:
   1. Create the **1****ldIr** directory in Assignment4. Pay careful attention to the name; every letter in the name is a different character (you should have a clear knowledge of how some characters look the same in Linux but are different).
   2. Under directory **1ldIr** create two new directories named **a0oO** and **bOo0** (four characters each, including one digit).
   3. Under directory **a0oO** create new directory **O0osub** (six characters, including one digit).
   4. Under directory **bOo0** create new directory **0Oosub** (six characters, including one digit).

**\*\* Have correct name, exactly same as specified in instructions. Failed to use the correct name leads to loose points.**

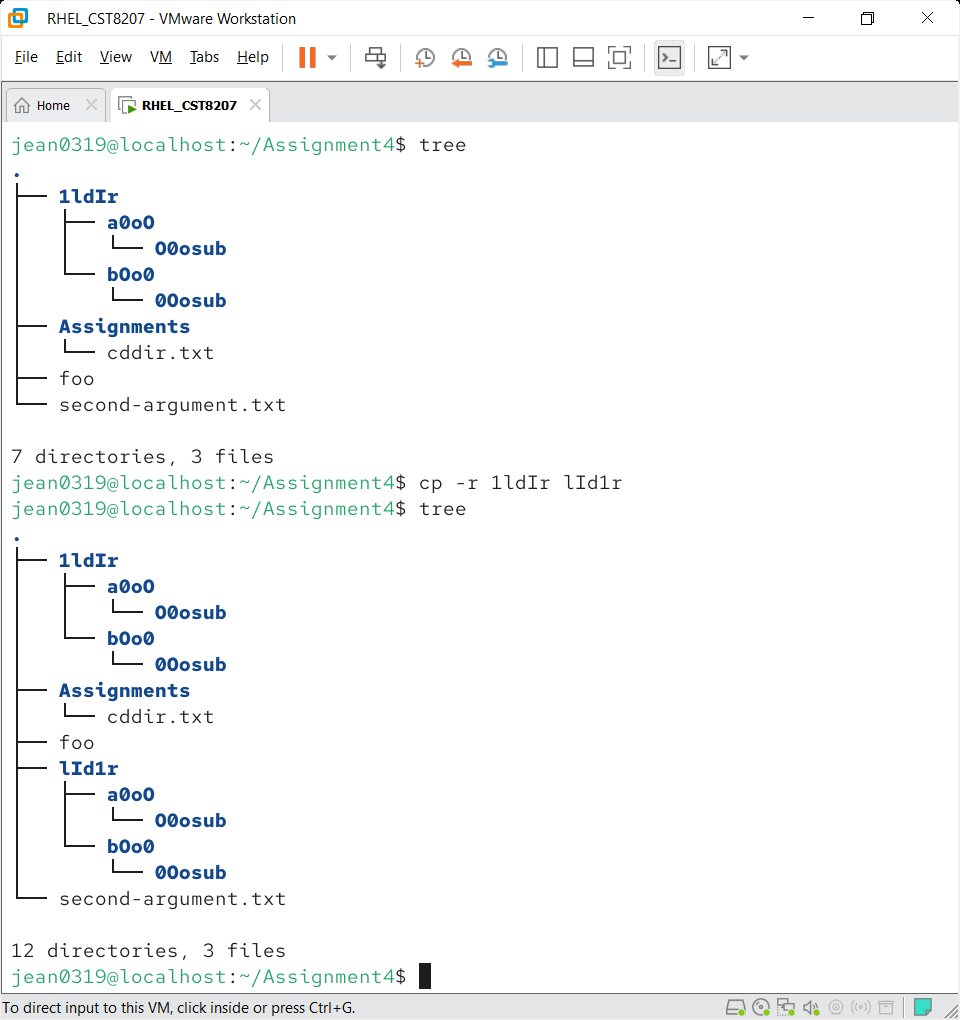


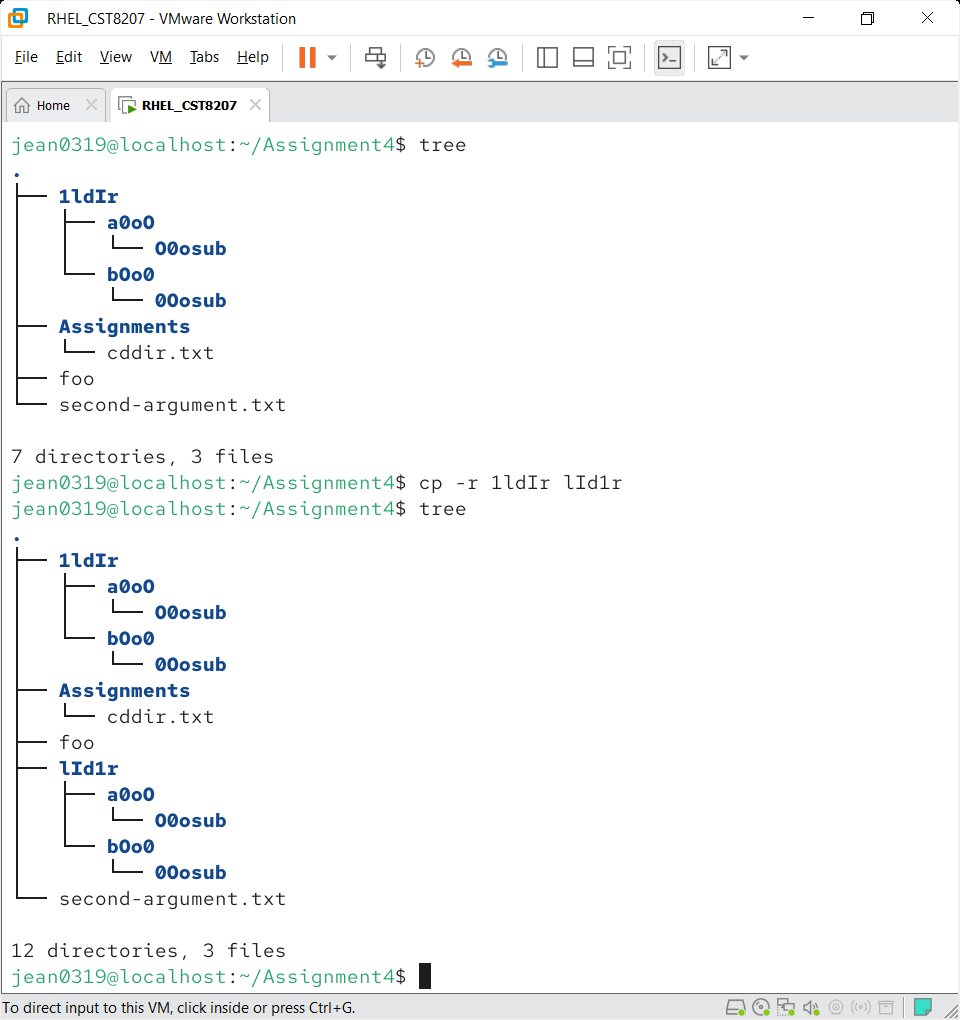
1. Copy just the content of directory **1ldIr** to the new directory **lId1r** using the correct directory copy option.

**Hint:** The new directory **lId1r** must not exist before you do this copy! If it already exists, recursively remove it before you do the copy, or else you will get a spurious incorrect extra level of directory such as lId1r/1ldIr

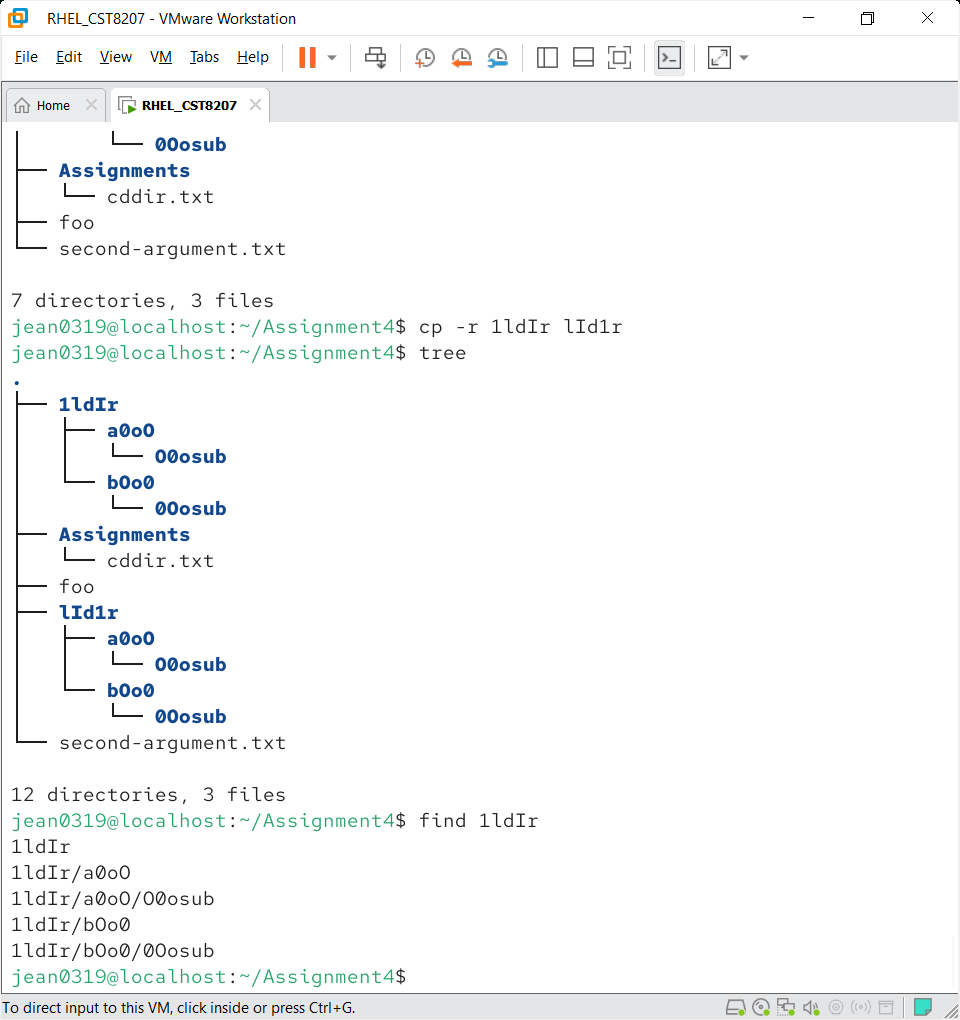
Check that the tree structure of **lId1r** is exactly the same as the tree structure of the **c** directory from which you copied it

**Add screenshot of command line**

****

****

**Execute tree command in your** Assignment4 **directory and add a screenshot of the output.**

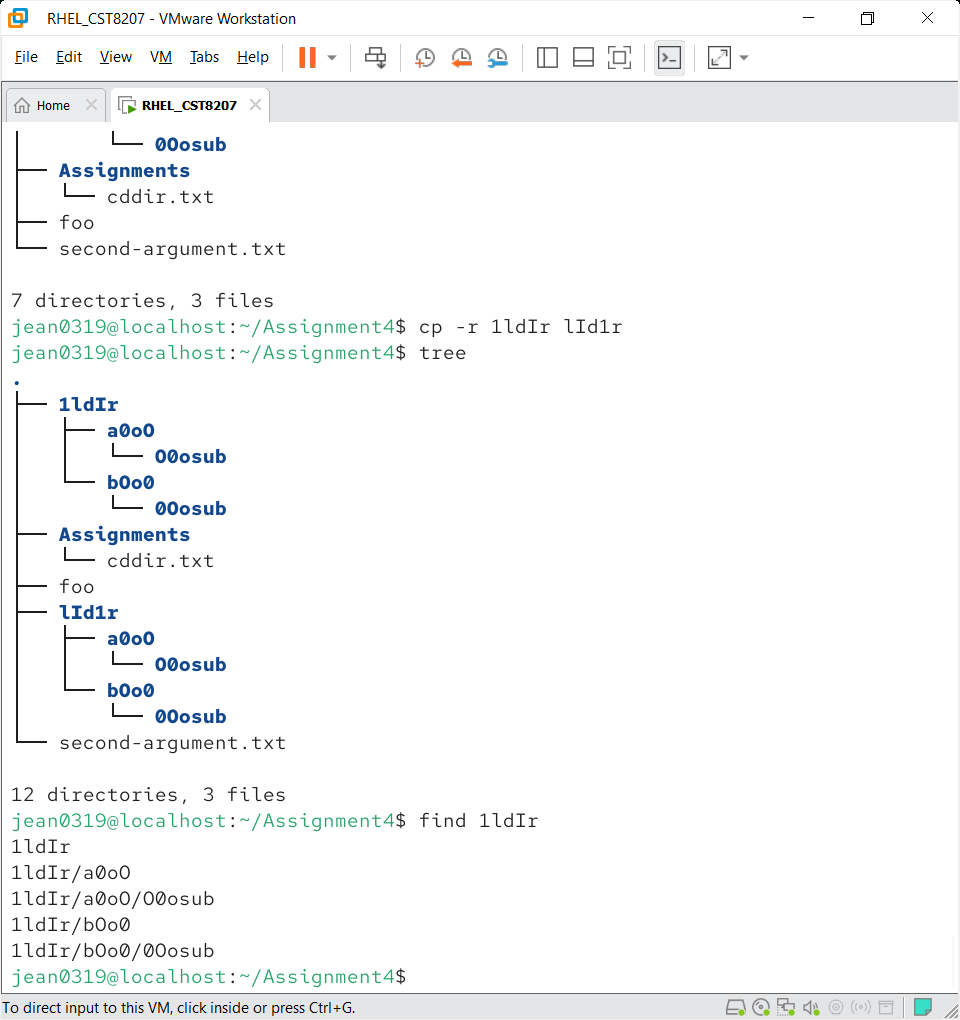
****

1. Use a command to recursively generate a list of all pathnames using your **lldIr** directory as the starting directory.

The recursive output of all pathnames under your 1ldIr directory will be exactly five lines long and **will contain this exact line in the output somewhere** (along with another four lines):

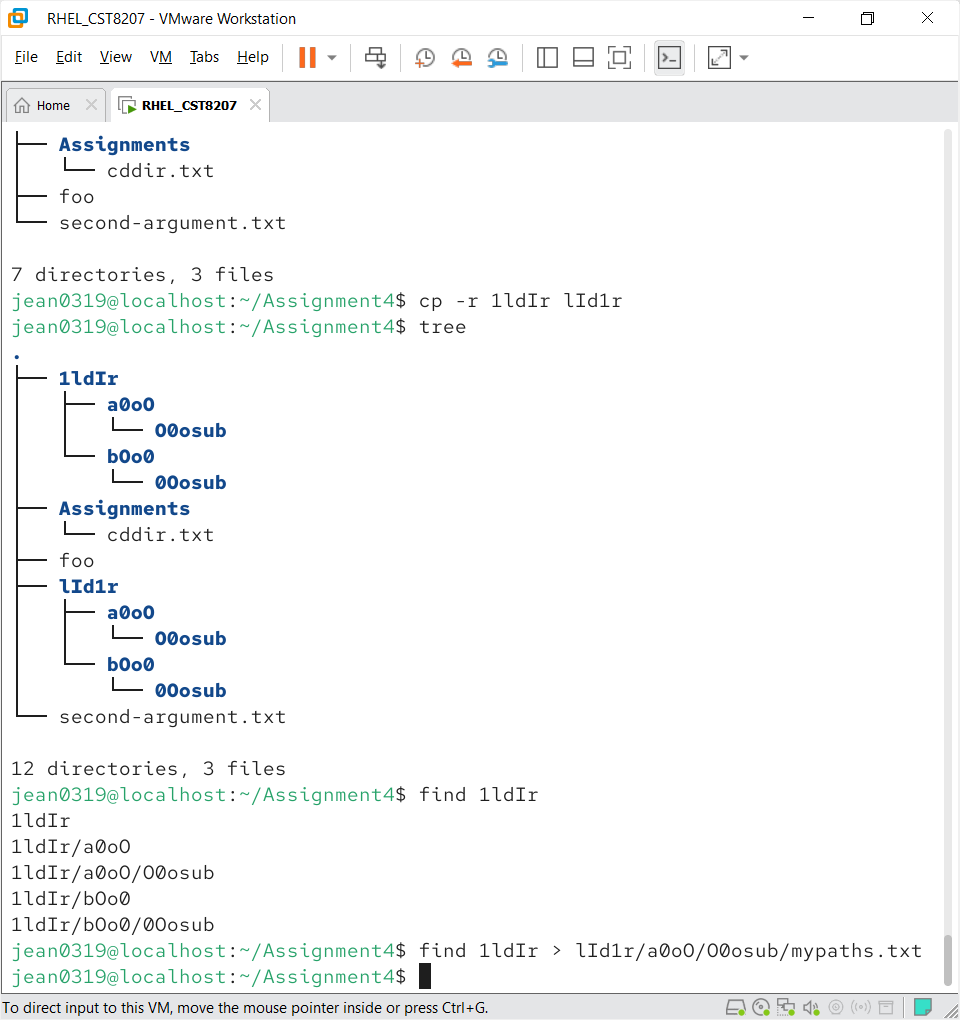


**Screenshot of the recursively generated list of all pathnames**

****

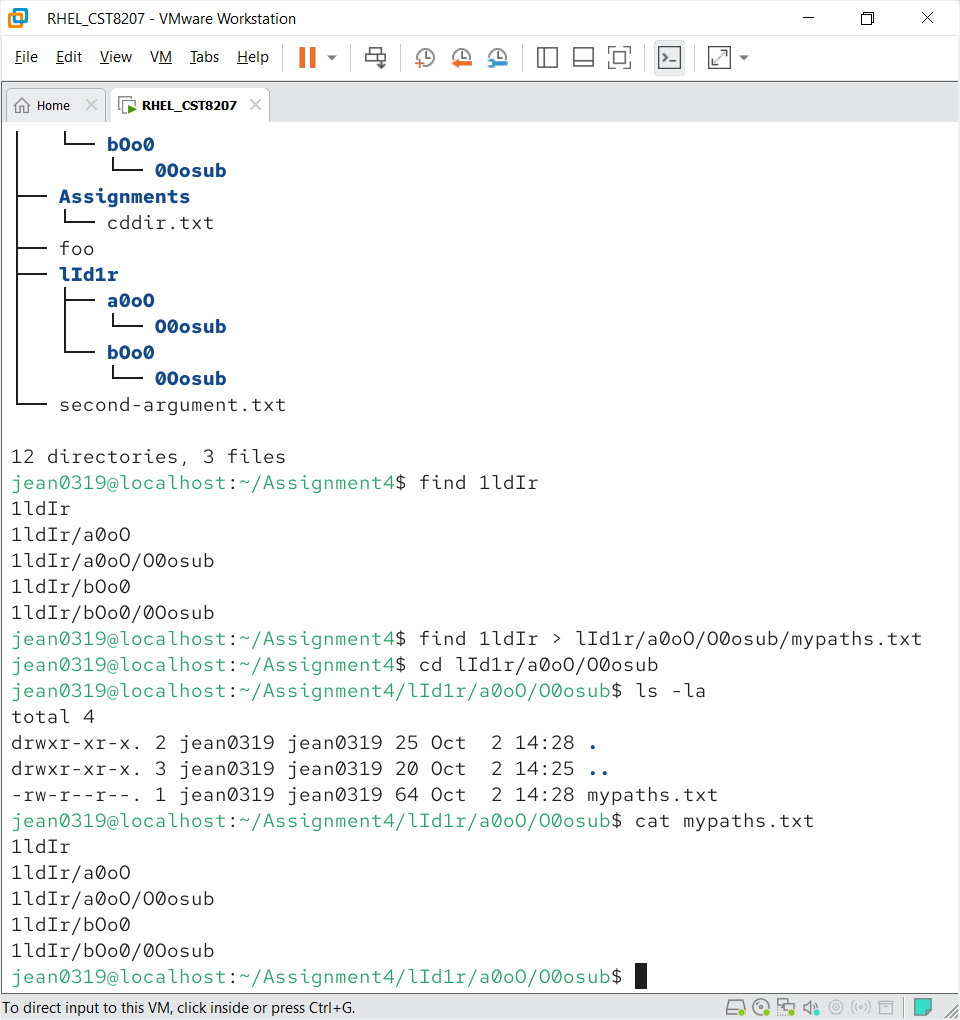
1. When the five lines of recursive output above on your screen looks correct, redirect the output of the command into the file named **myPaths.txt** in the **O0osub** directory that you created earlier under the **lId1r** directory **(not `1ldIr!).**

**Screenshot of command line**

****

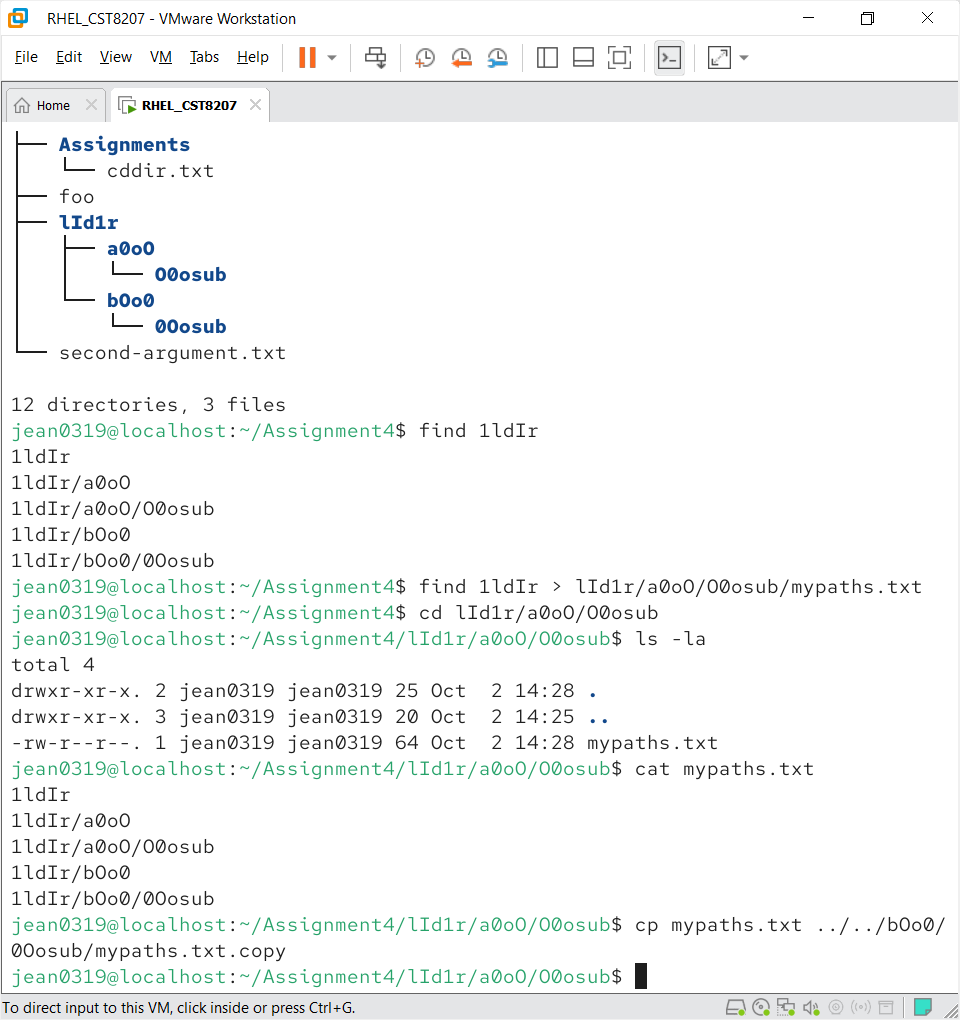
1. Use a command to list the files in the current (O0osub) directory to confirm that the myPaths.txt file is here. Use another command to display the contents of the myPaths.txt file, and make sure it contains exactly five lines of pathnames.

Screenshot of the content of file mypaths.txtfile



From in the O0osub directory, copy the myPaths.txt file (from the current directory) into the 0Oosub directory that is also under the **lId1r** directory, giving it the new name myPaths.txt.copy as the destination file name for the copy. Use relative pathnames to do the copy.

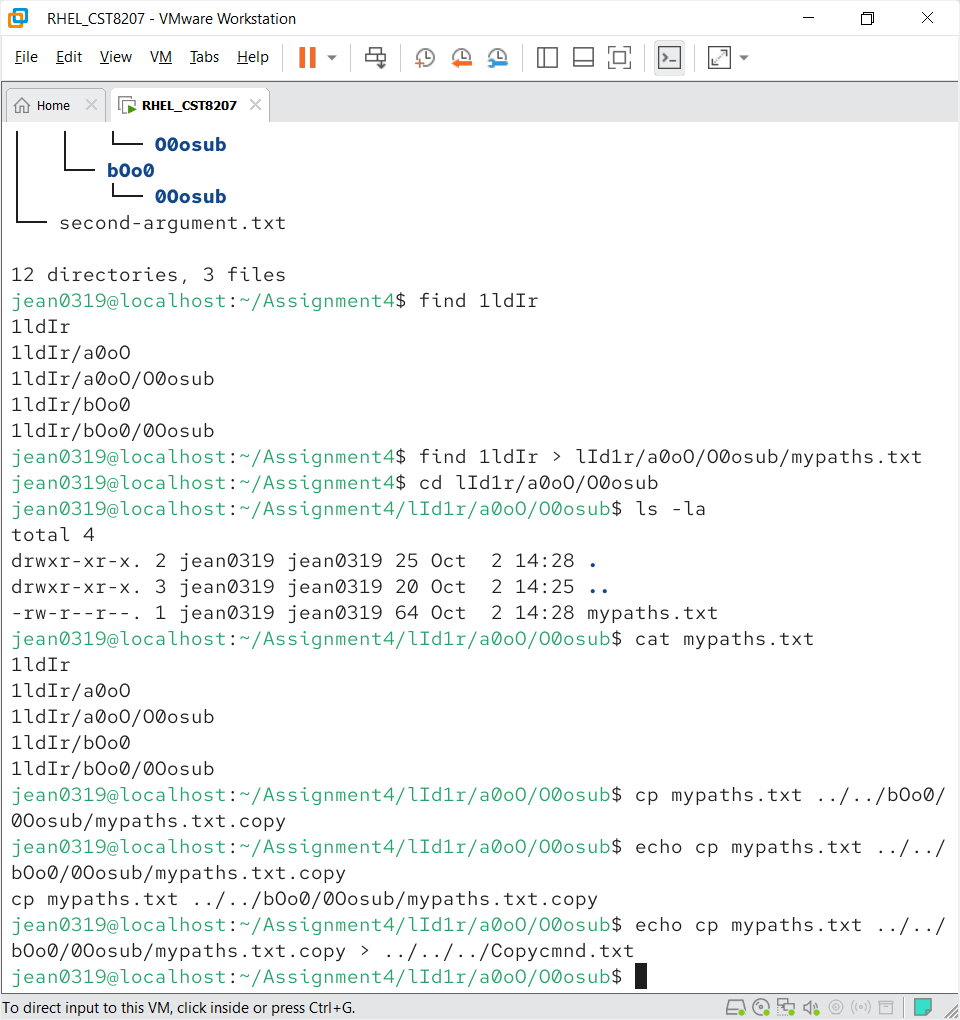
**Screenshot of the command line**

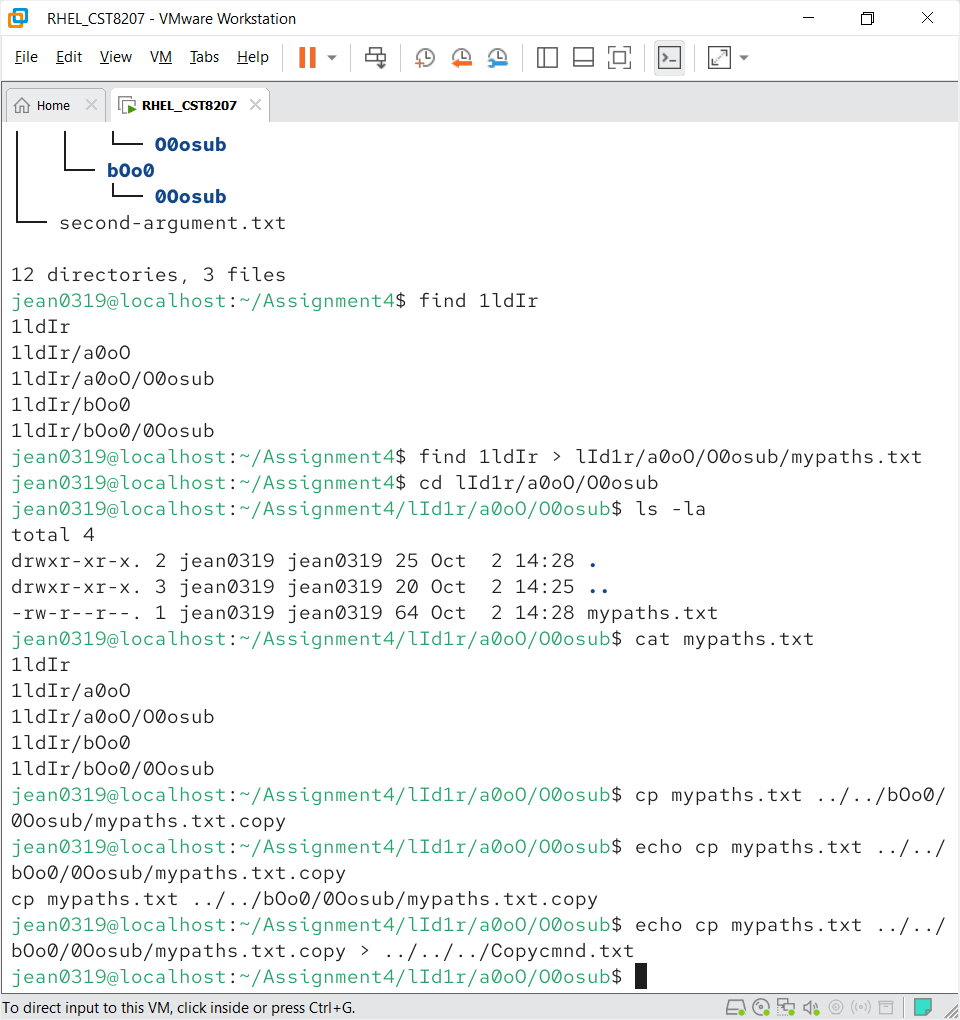
****

1. Save a copy of the command line you just used into a file as follows:
2. Use the Up-Arrow key to retrieve the copy command you just typed.
3. Use the Left-Arrow key to move all the way left and then prefix the copy command line you just typed with the word echo; type the word echo followed by a space at the start of the command line, in front of the copy command name.
4. Push the [Enter] key and confirm that the copy command name and its two relative pathname arguments echo onto the screen.
5. Again use the Up-Arrow key to retrieve the echo command you just typed.
6. To the right end of the echo command line, add redirection to redirect the echo output into file Copycmnd.txt in the Assignment04 directory. No output should appear on screen.

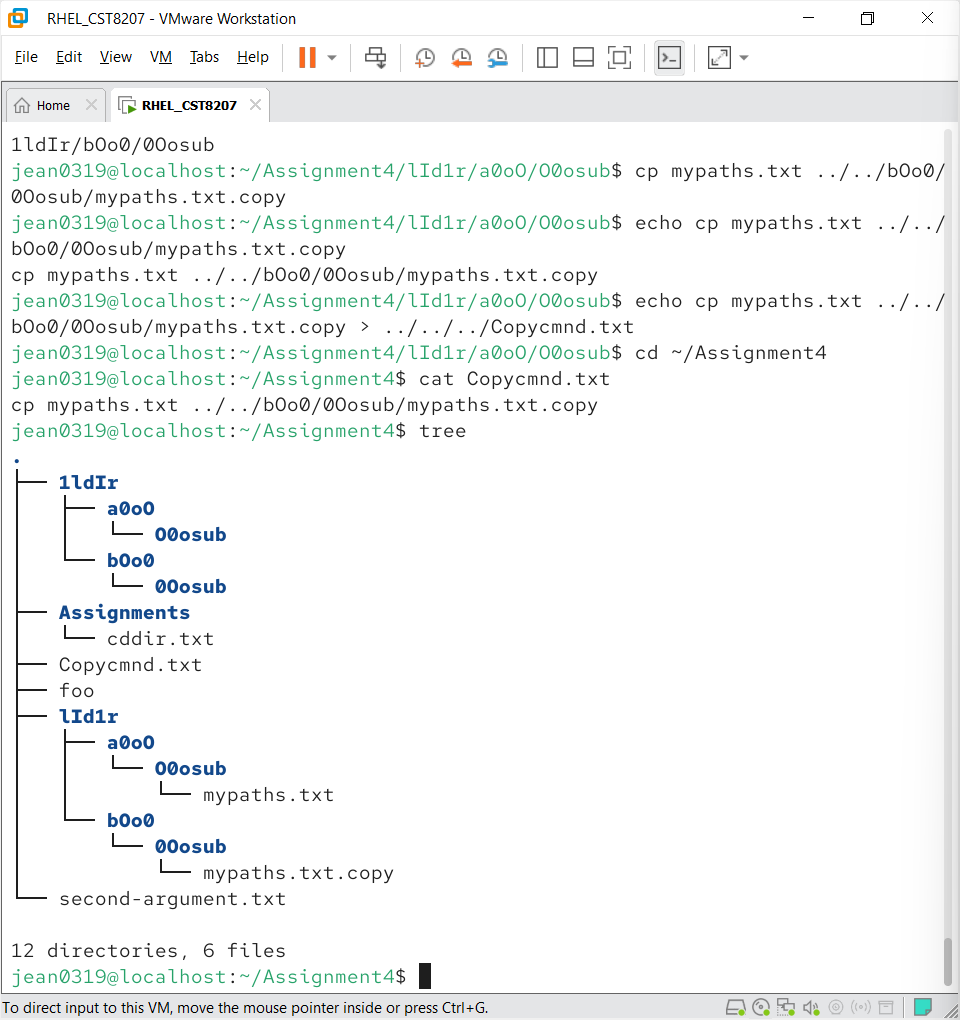
The Copycmnd.txt file should contain on one line: the copy command name followed by two relative pathname arguments, exactly as you typed it in the preceding question.

Screenshot of command line





Screenshot the content of file Copycmnd.txt



**Task 4: cd, mkdir, touch, mv, rm, cp, mkdir, find**

Enter exactly the commands that are shown in bold below and note which commands produce errors. (There will be three errors; this is intentional.) Answer the questions following based only on these review commands in execution.

Be precise in your typing!

|  |  |
| --- | --- |
| 1. **cd ; rm -rf ~/lab4.8** | 9. **cp tomato lettuce garden** |
| 2. **mkdir ~/lab4.8** | 10. **mkdir jardin forest** |
| 3. **cd ~/lab4.8** | 11. **mkdir garden/flower** |
| 4. **mkdir ./orchard** | 12. **rmdir jardin** |
| 5. **touch apple orange** | 13. **touch lab4** |
| 6. **mv orange orchard/lemon** | 14. **cd orchard** |
| 7. **rm orange** | 15. **cd ../../lab4.8/forest** |
| 8. **touch lettuce tomato cucumber** | 16. **mv ../lab4 ../tomato** |

1. Give the command number that generated the error followed by the full and exact error message:

rm orange

command 7error: rm cannot remove orange

command 9

command 11

1. What is the absolute path of the shell's current working directory after the last command (16), above?

Ans: /home/jean0319/lab4.8/forest

1. Give the absolute pathname of the one regular file lemon that is now in the directory named orchard:

Ans: /home/jean0319/lab4.8/orchard/lemon

1. Give the relative path to the same lemon file from the forest directory:

Ans: ../orchard/lemon

1. Give the relative path to the same lemon file from your own HOME directory:`

Ans:

lab4.8/orchard/lemon

1. Give the relative path to the same lemon file from the directory called /home:

Ans: jean0319/lab4.8/orchard/lemon

1. Give the relative path to the same lemon file from the Linux ROOT directory:

Ans:

home/jean0319/lab4.8/orchard/lemon

1. Give the relative path to the same lemon file from the directory called /root:

Ans:

../home/jean0319/lab4.8/orchard/lemon

1. List the basenames of directories that were successfully created (at any time) during the review exercise:

Ans: successfully created lab, forest, lab4.8

1. List the absolute pathnames of the five regular files still remaining anywhere under the directory lab4.8. Do not include the names of any directories or sub-directories – list only the five absolute regular file names located anywhere under the review directory lab4.8:

Ans:

- ~/lab4.8/apple

- ~/lab4.8/cucumber

- ~/lab4.8/lettuce

- ~/lab4.8/orchard/lemon

- ~/lab4.8/tomato

1. What command line recursively finds and displays all pathnames under ~/lab4.8 ?

Ans: find ls ~/lab4.8

1. What does the option **-ls** do when used with **find** command? (See "man find".)

Ans: the ls option list option list directory contents

1. What command line recursively finds and displays only pathnames ending in "log" in the system directory /etc (you will see many permission denied messages in the output, as well as pathnames)?

Ans: find /etc -name “\*.log”

1. What does the find expression "-type f" do? (See "man find".)

Ans: -type f restricts the search to only files

**Task 5**

1. Can you write below command line in a more compact form? If no explain why ? if yes write it



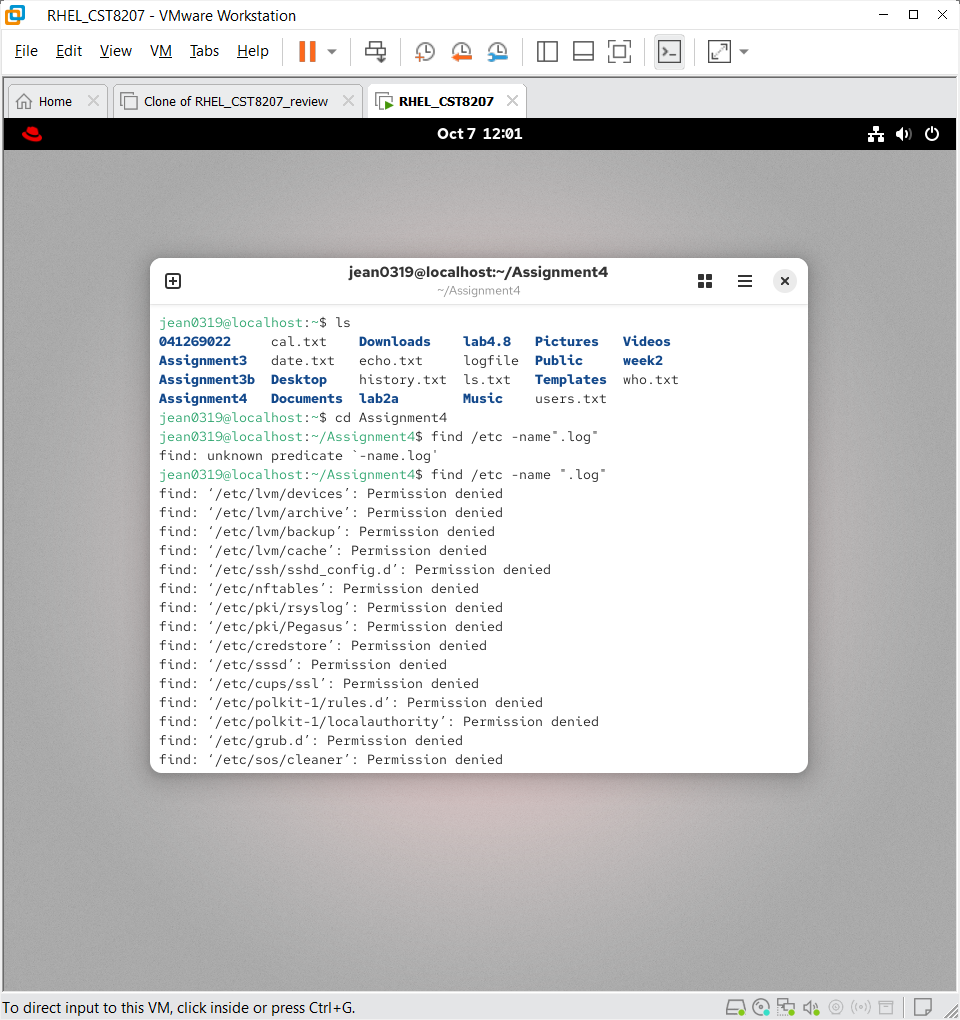
Ans- grep john file1 > result

* + 1. What does the option **-ls** do when used with **find** command? (See "man find".)

Ans: the -ls option prints detailed information about each file

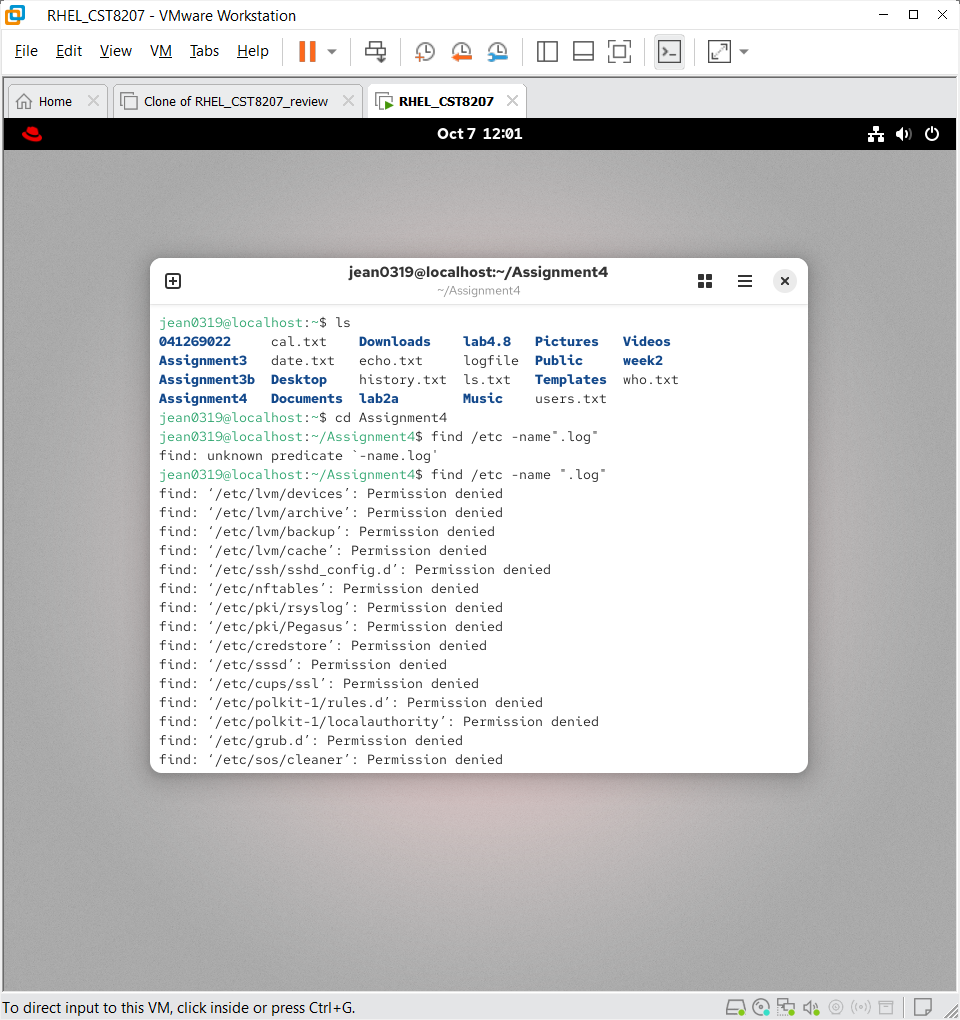
* + 1. What command line recursively finds and displays only pathnames ending in "log" in the system directory /etc (you will see many permission denied messages in the output, as well as pathnames)? Write command line and execute

Ans: screenshot of command line   
  
find /etc -name “\*.log”



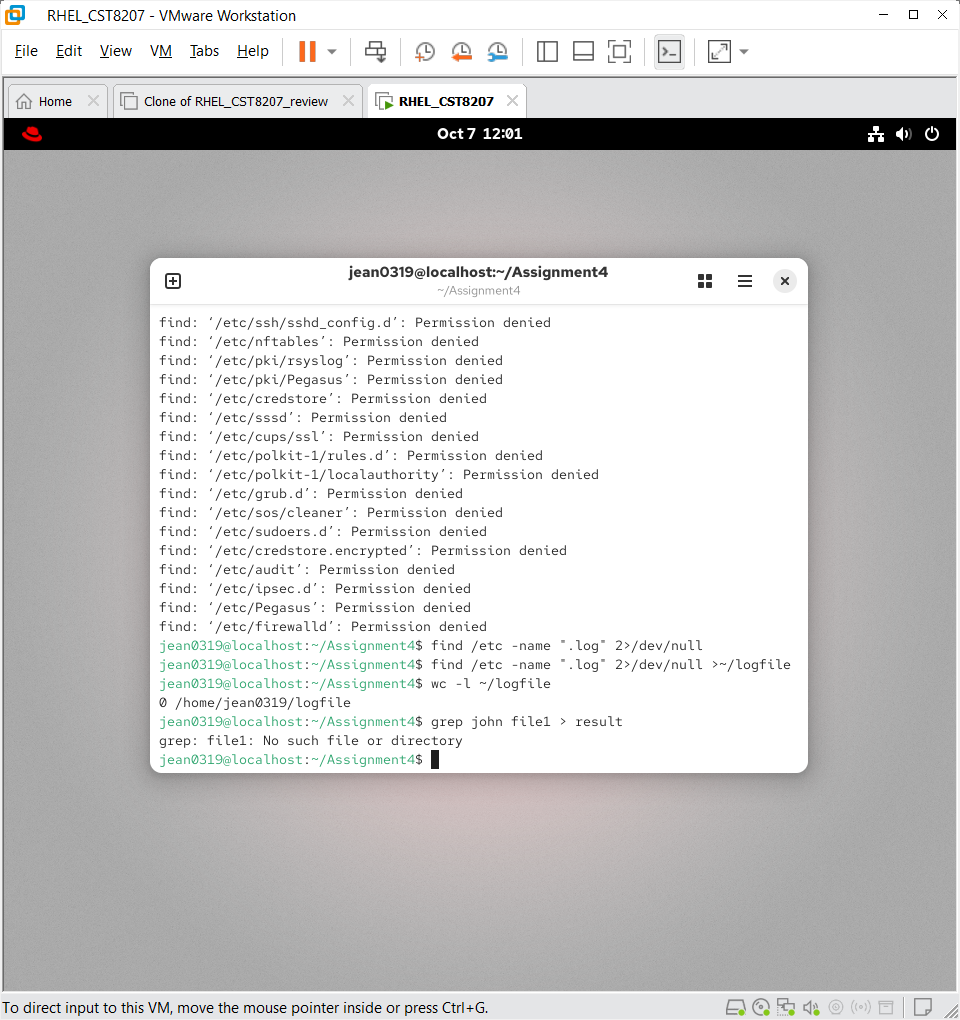
* + 1. Now modify the above command line (3) so the output just shows the pathnames, no error messages should be displayed on terminal

Ans: screenshot of command line



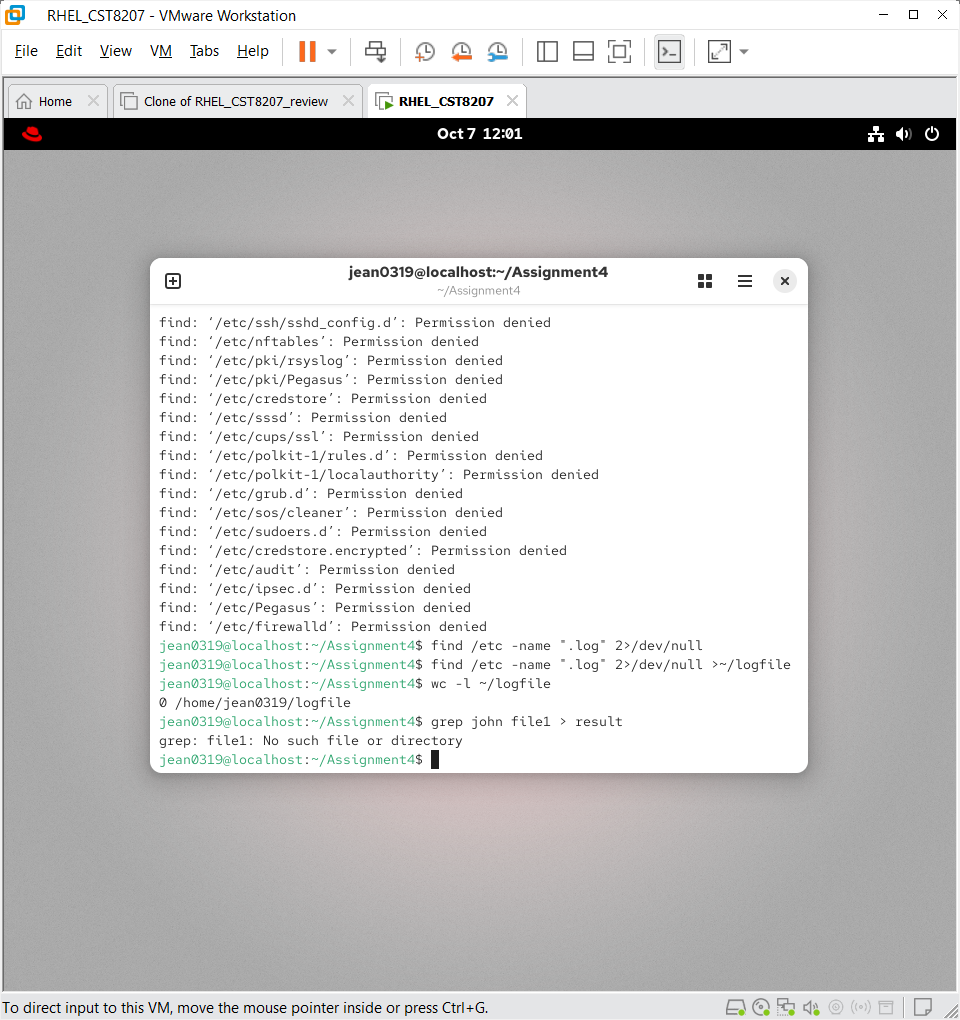
* + 1. Now further modify the command line to save all pathnames generated above to file ~/logfile

Ans: screenshot of command line



* + 1. Now count how many pathnames are in file ~/logfile

Ans: screenshot of command line



**\*\*\*After completing your assignment demo your work to your lab professors.**