Questions and Exercises to work out and turn in:

Grading Guidelines (See Appendix):

In general, a right answer will get full credit when:

1. It is right (worth 25%)
2. It is right **AND** neatly presented making it easy and pleasant to read. (worth an **extra** 15%)
3. There is an **obvious and clear link[[1]](#footnote-1)** between 1) the information provided in the exercise and in class and 2) the final answer. A clear link is built by properly writing, justifying, and documenting an answer (worth an **extra** 60%).
4. Calculation mistakes will be minimally penalized (2 to 5% of full credit) while errors on units will be more heavily penalized.

**Late Submission** : as specified in the syllabus. Day counting starts one minute after the deadline.

**Check Your Submission:**  after submitting, download your submission to check whether it is the right version and it is complete. after submitting, download your submission to check whether it is the right version and it is complete. A wrong version will get at least a 25 points penalty that cannot be claimed back by the *Leniency Policy for Borderline Incidents*. So, check your submissions.

* You are welcome/encouraged to discuss exercises with other groups or the instructor. But, ultimately, **personal** writing is expected.
* USE THIS EDITABLE FILE AS THE STARTING DOCUMENT YOU WILL TURN IN. KEEP IN THE QUESTIONS AND INSERT YOUR ANSWERS RIGHT AFTER THE QUESTIONS/PROMPTS. DO NOT DELETE ANYTHING.
* IF USING HAND WRITING (STRONGLY DISCOURAGED), REWRITE THE QUESTIONS.
* FAILING TO FOLLOW TURN IN DIRECTIONS /GUIDELINES WILL COST A 30% PENALTY.

Objectives of this assignment:

* Resume introduction to basic Unix commands.
* Explore commands related to processes

What you need to do:

* Execute and understand basic Unix commands on Engineering Tux machines. *You must complete all tasks on an Engineering Unix Tux machine, otherwise a 0 will be assigned.*
* Always document your work by taking/inserting screenshots. To save space, clip out the screenshots to contain only relevant information. See Appendix for the required information.
* Insert answers as indicated (right after the question/directions)

(Reminder) **In order to save space, for this assignment and future ones, clip out the screenshots to contain only the relevant information (*the date, the Tux machine you are using for the question, the Auburn username, and material related to the question/task*). A 25 points penalty will be applied when any required information is missing. Make sure that the screenshots are easily readable.**

Exercise 1: Basic Unix Commands (30 points)

(Well written short answers are acceptable for this assignment)

The objective of this exercise is to get familiar with basic frequently used commands.

**Task**:

Consider these basic frequently used Unix commands:

who, history, pwd, cp, mv, kill, date, cat, ps, top

For each of the above commands,

1) Provide a brief description (you may use the **man** command, but use ultimately your own words)

The command who pulls up the list of users that are logged into the system.

The command history pulls up a list of past executed commands in the terminal.

The command pwd stands for print working directory and it shows the current directory that is working.

The command cp copies both files and directories to a different location.

The command mv allows to move or rename files and directories.

The command kill terminates a running process by entering its Process ID.

The command date shows the time and date on the system.

The command cat both displays the contents of the state file and combines files.

The command ps shows the process that are running at that time.

The command top shows the usage of CPU and memory in real time.

2) Execute the command on a Tux machine

3) Briefly comment the results/outcomes of the execution

The command shows that I am currently logged into the tux machine.

The command history shows all of the past commands that I have entered including clear and date.

The command pwd shows /home/u3/kem0149.

The command cp shows cp: missing file operand because I haven’t entered a file to copy, but when I do it shows that it was successfully moved.

The command mv when I select lab2.c and 3500 it renames the file lab2.c to 3500. The command kill brings up the screen of an example of a process and how to kill it.

The command date shows the date and time that I am on the machine.

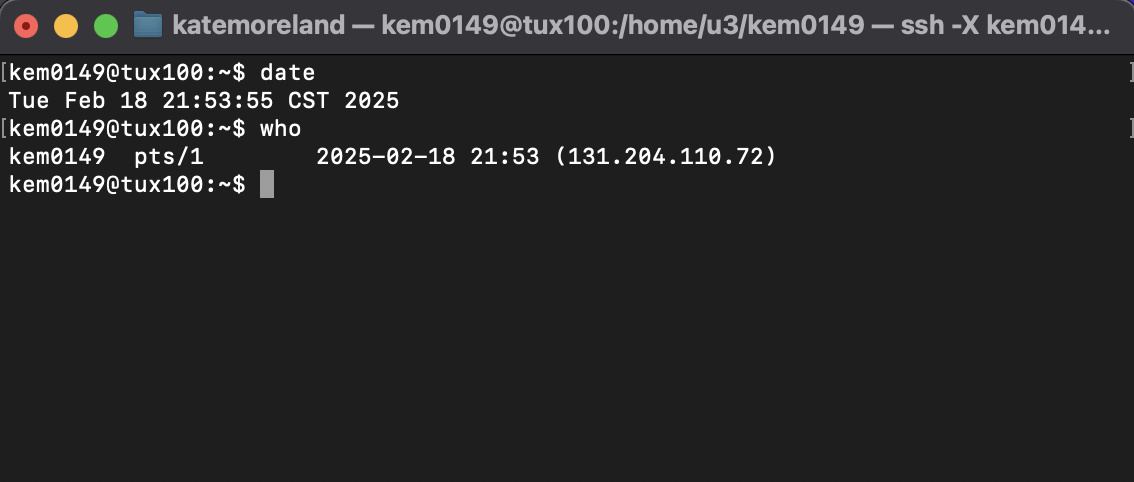
The command cat followed by lab2 shows that lab2 is in the file.

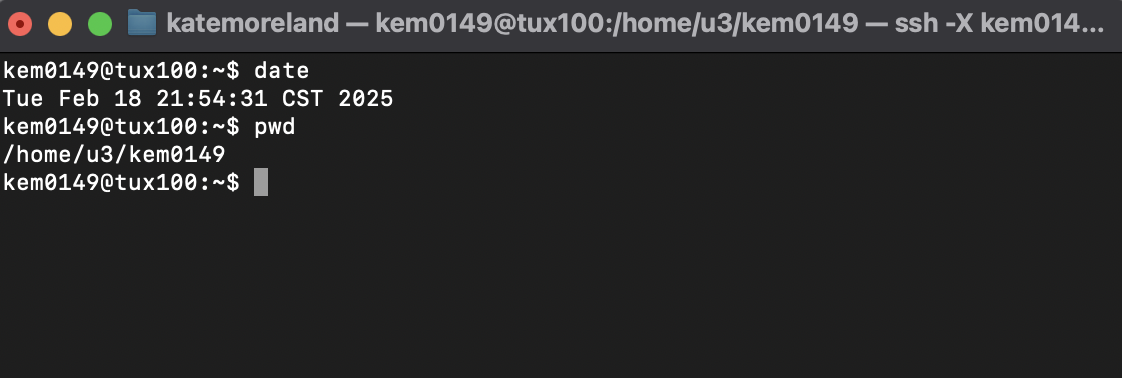
The command ps shows the bash and ps command.

The command top brings up what all is taking up my CPU and memory on the tux machine.

4) Report any unexpected behavior.

Provide a screenshot only for the commands who and pwd.





Exercise 2: Process Management Tools/Commands (70 points)

The objective of this exercise is to observe the "life" of processes using two commonly used Unix commands: ps and top.

**In order to produce answers consistent with the answer key, you must execute ONLY the commands you are asked to perform in exactly the order given.**

In order to complete this hands-on lab, you must complete the following actions:

- **Download** the C program lab2.c from Canvas

- **Move** lab2.c to your Auburn home directory lab2 that you created in M1: hands-on laboratory exercise. This directory's path is ~/nnnn/lab2 where ~ is your Auburn home directory path (watch if necessary the video about how to move files from your local machine to your Auburn home directory). Observe that you do not need to know the C programming language to complete this hands-on laboratory exercise. When executed, the program lab2.c creates (forks) three child processes. The parent process and children run infinite loops. The only way to stop them is to kill them.

- **Log on** any Tux machine

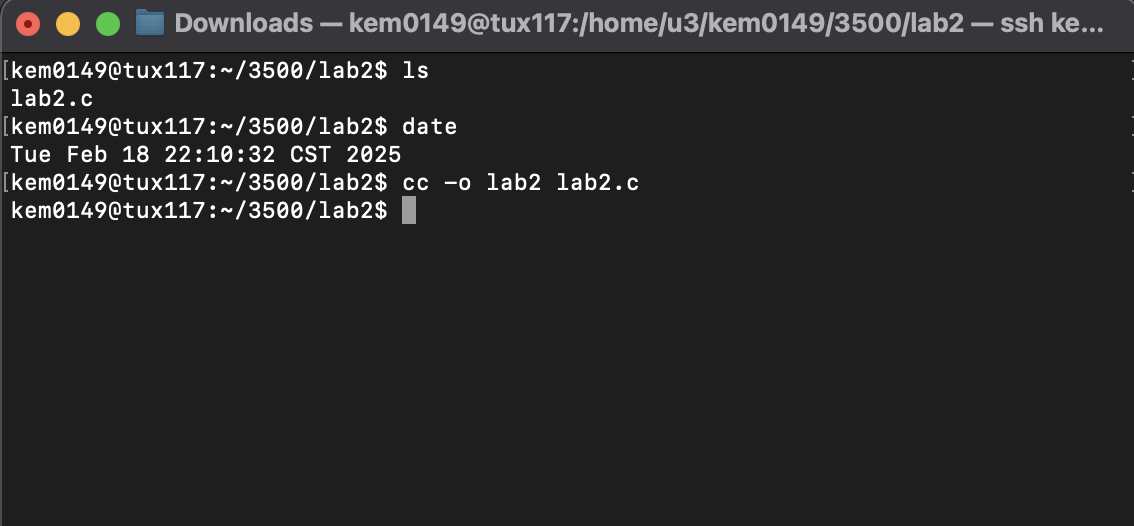
- **Go** to your lab2 directory at ~/nnnn/lab2 (See M1: Hands-on Lab..)

- **Type** **ls** to insure that the file *lab2.c* is in the current directory (i.e., ~/nnnn/lab2)

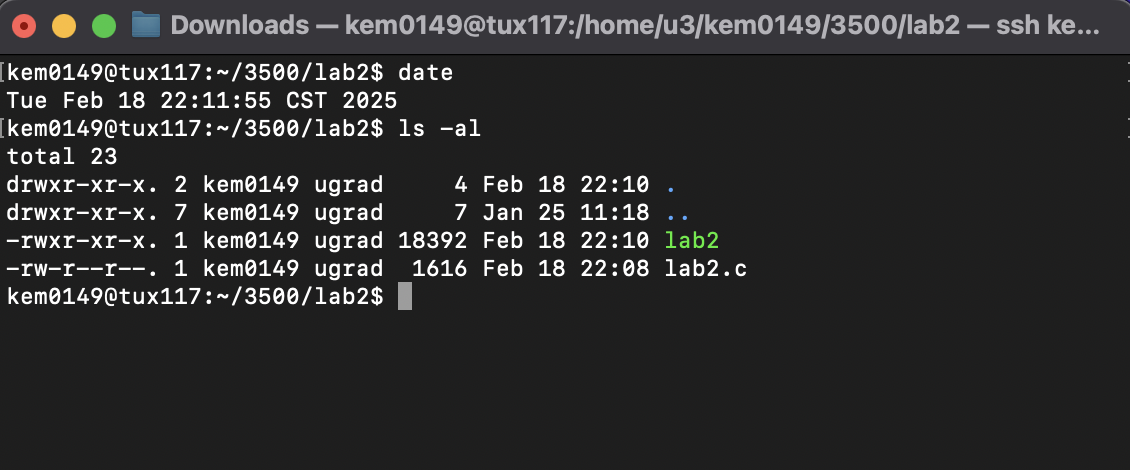
- Type **date** .

- **Type** the command **cc -o lab2 lab2.c** (This will compile lab2.c and produce executable lab2)

- **take and insert here a screenshot (clip out …..)**



- **Insure** that the compilation is successful. Check whether the executable lab2 is in the current directory (use ls -al). If successful, complete the following actions and answer the questions:



1) (7 points) Type the **ps** command (consider a man ps to learn about ps)

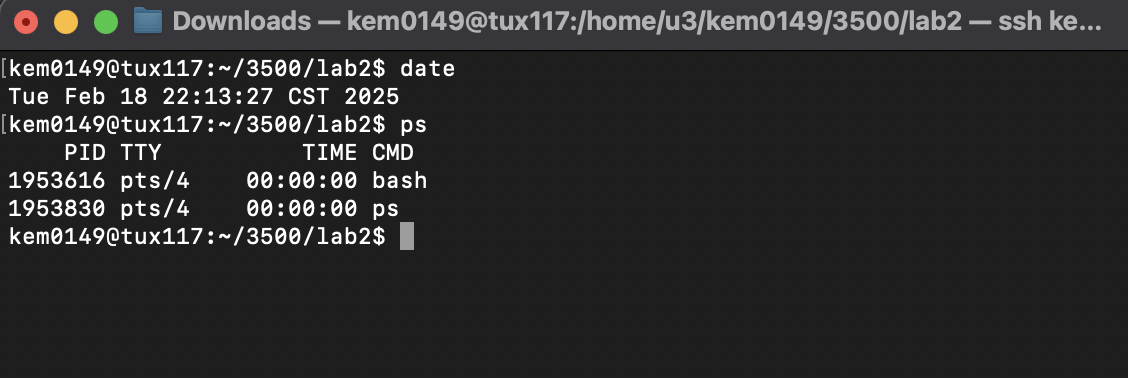
What are the processes IDs and names listed by the command ps?

The process IDs and names are:

1953616 – and the name is bash

1953830 – and the name is ps

Insert a screenshot here to document your answer



2) (7 points) **Type** the command **./lab2&** followed by return and then **type** the command **ps**. Do not forget the '**&**' character. If you do not add the '**&**' caracter, you will not be able to get the prompt for the next commands.

What are the processes IDs and names listed by the command **ps**? You can insert a screenshot here to answer the question

1953616 – the name is bash

1953861 – lab2 Parent Process

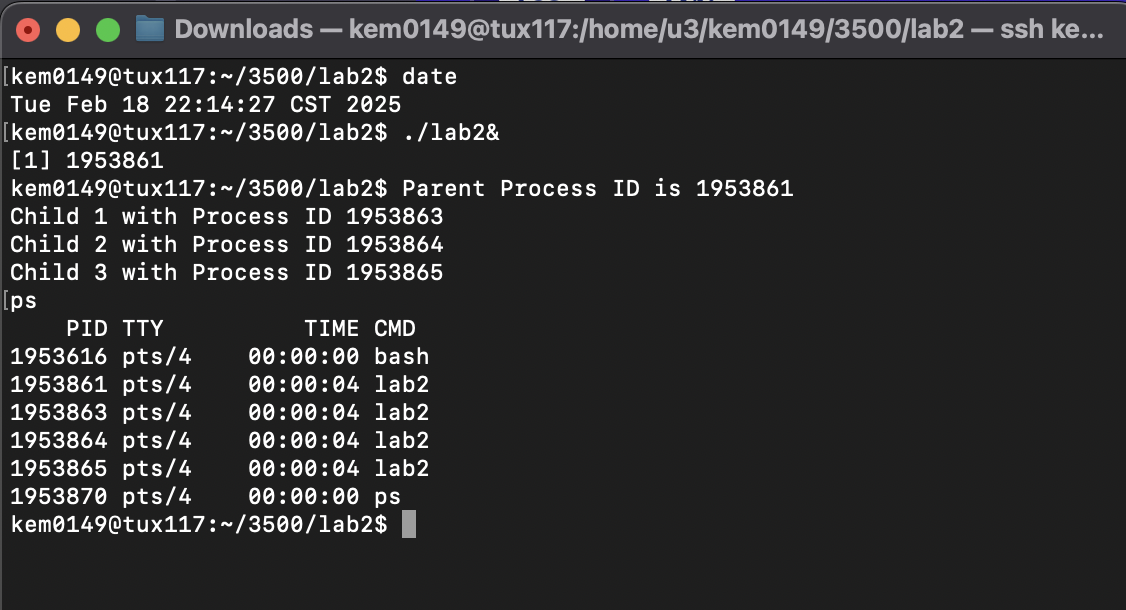
1953863 – lab2 Child 1

1953864 – lab2 Child 2

1953865 – lab2 Child 3

1953870 – the name is ps

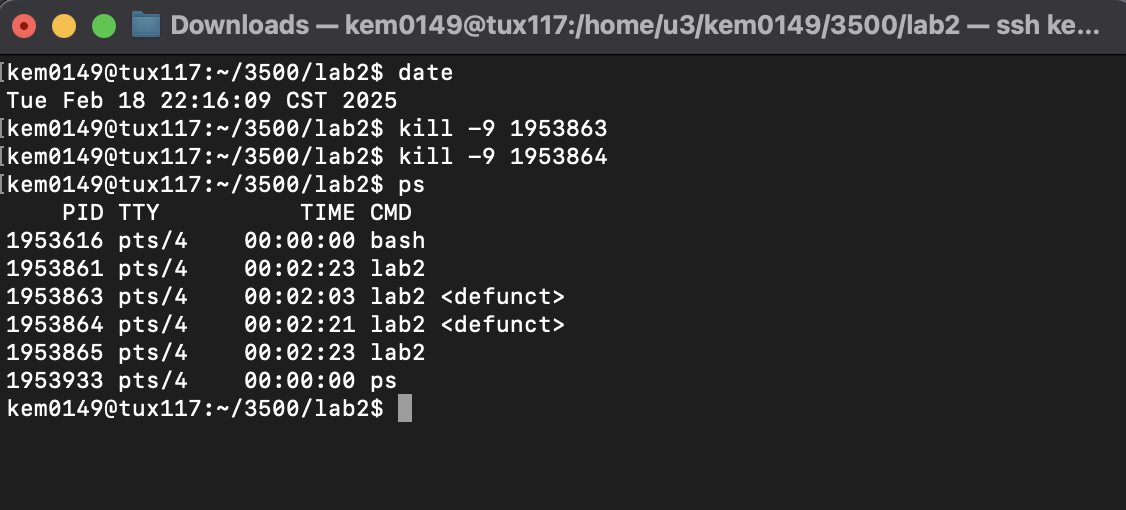
Insert a screenshot here to document your answer (clip out ….)



The program lab2 creates 4 processes: a parent process and 3 child processes. The parent has a smaller process ID than its children's.

3) (7 points) In order to kill a process with process ID *n*, you must type the command kill -9 n. For this question, kill two **child** processes. Type the command **ps** and take a screenshot

Insert the screenshot here...... (clip out the screenshot to show only the relevant information)



Do the killed processes show up on the screenshot?

Yes, the processes still show up.

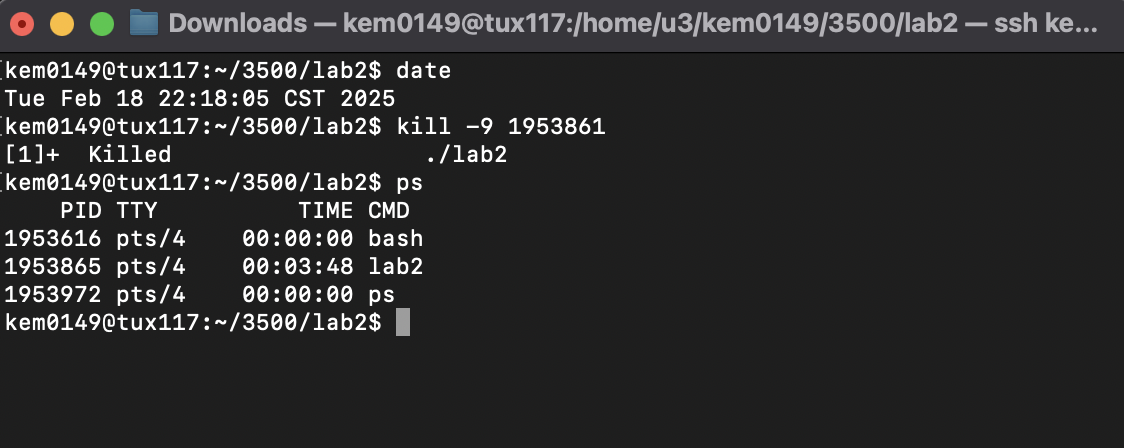
If the two child processes still show up, how are they different from the alive child process?

They are labeled defunct, which means that the child process is finished but the parent hasn’t seen that it should be exited yet. It means that it is dead but it is still shown after ps.

4) (7 points) Kill the parent process and type the **ps** command

Take a screenshot

Insert the screenshot here...... (clip out the screenshot to show only the relevant information)



Do the killed processes show up?

No, the killed processes do not show up after the ps command.

Does the alive child process still show up?

Yes, the alive child still shows up.

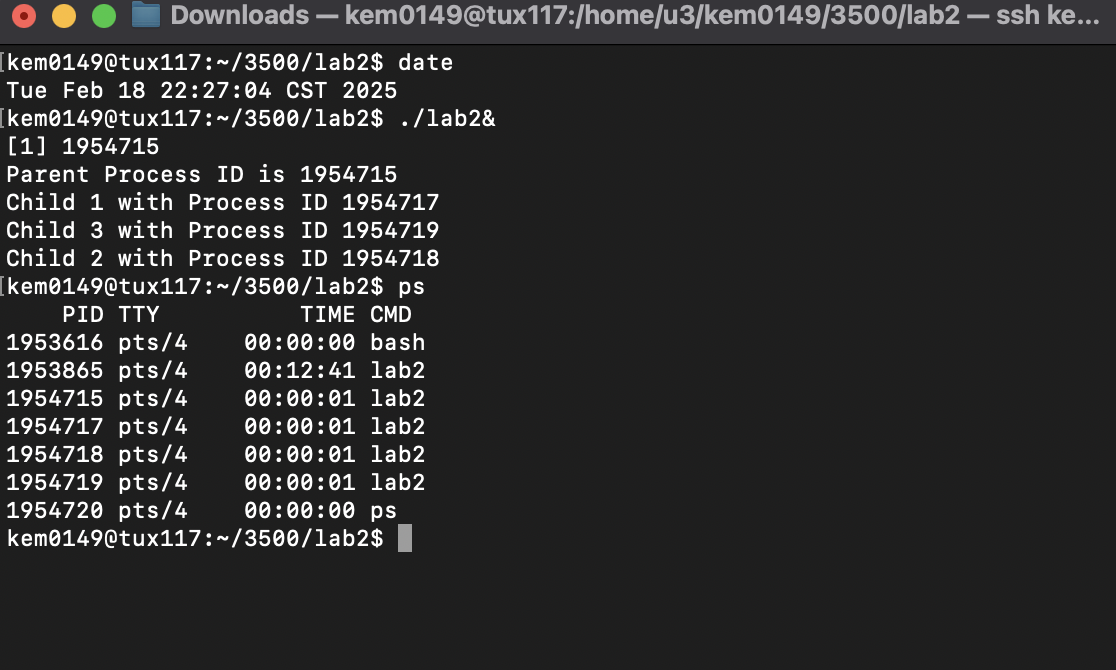
If yes, kill it. Check with **ps** that all processes created by lab2 were killed.

5) (7 points) Let us call ***Terminal A*** the terminal you used so far for all the previous questions you answered so far. Keep Terminal A open and open a new ***terminal B***. Use Terminal B to log on the **SAME** Tux machine you used under Terminal A. Execute on Terminal A the command **./lab2&** followed by return and then the command **ps.**

Do you see (on Terminal A) the processes created by the program lab2?

Yes, I can see the process that was created by lab2 and the children of the processes.

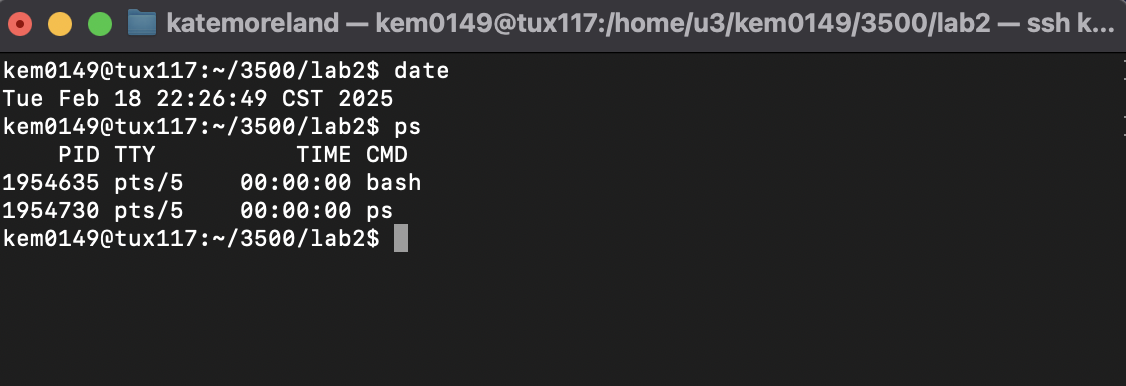
Insert a screenshot here to document your answer



Type on Terminal B the command **ps**. Do you see (on Terminal B) the processes created by the program lab2?

No, on Terminal B I am not able to see the processes that are created on Terminal A.

Insert a screenshot here to document your answer



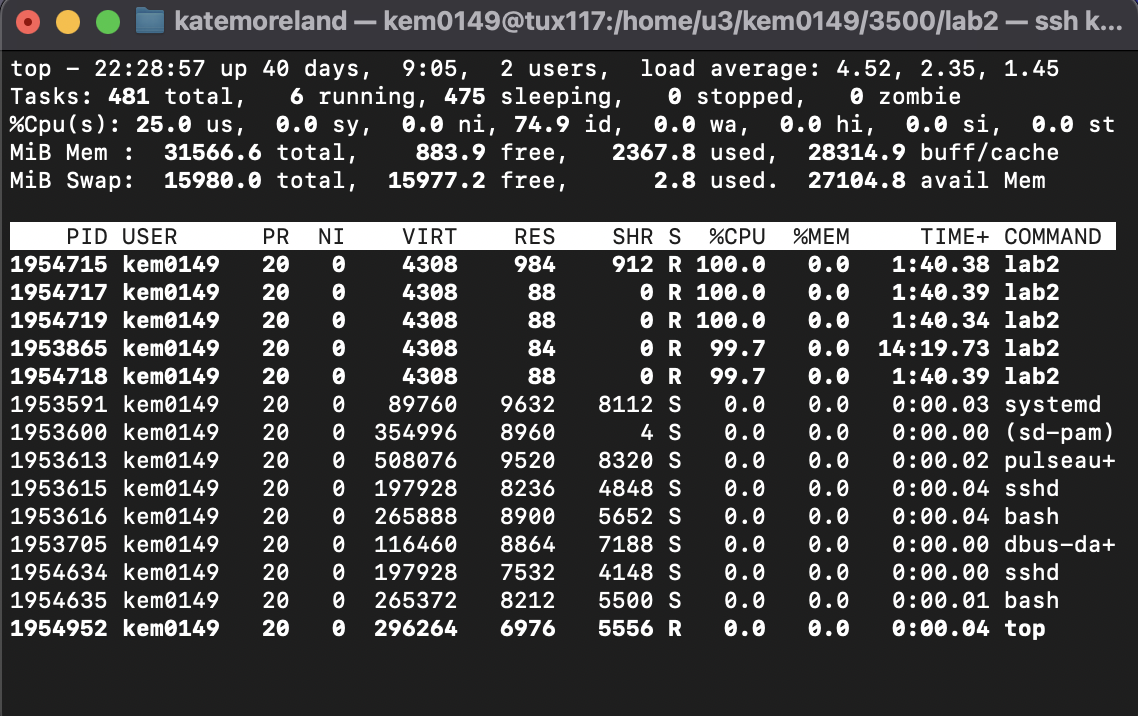
If no, can you explain why the processes created by the program lab2 do not appear on Terminal B when executing **ps**? Do a **man** **ps** or Google **ps** to understand and document your answer.

Each terminal is assigned a different identifier. If you do not clarify which terminal after the command ps then it will just show the current terminal’s processes.

6) (7 points) On Terminal B, type the command **top -U username** (where username is your Auburn username).

Do you see on Terminal B the processes created by the program **lab2**?

Yes, you can see the process created by lab2 on the terminal B.



How many "zombies" appear on Terminal B? (Search for "Zombie" on Terminal B...)

There are 0 zombies that appear on Terminal B.

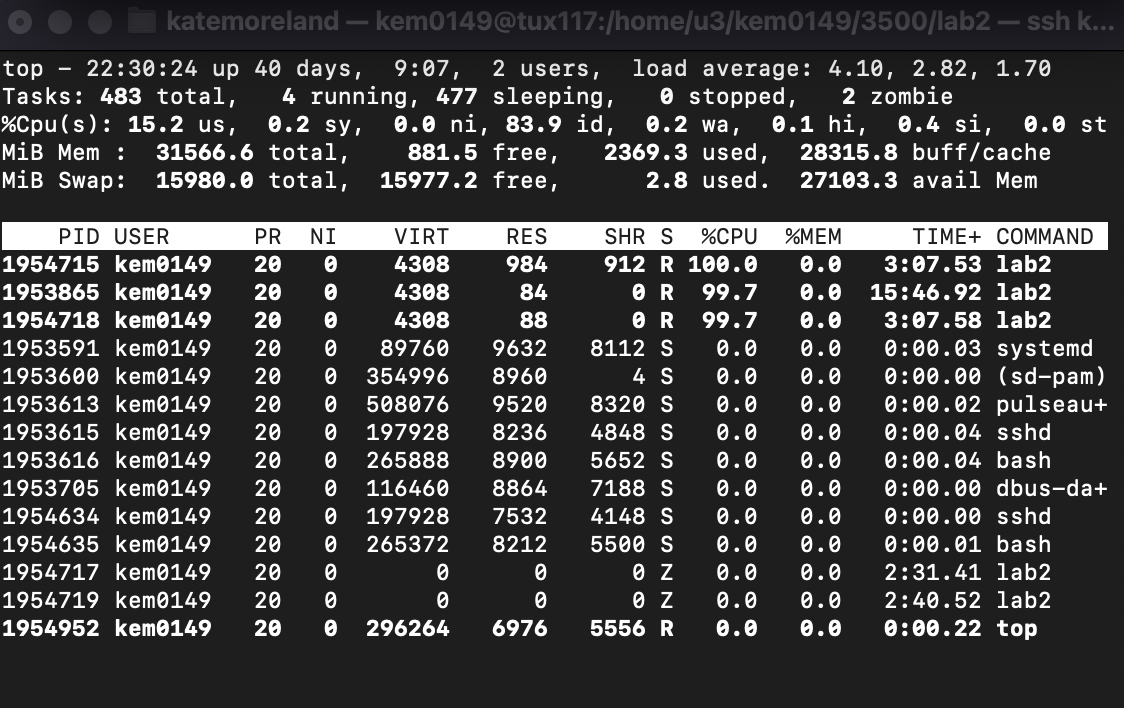
7) (7 points) On Terminal A, kill two child processes.

Do you see on Terminal B the processes created by the program lab2?

Yes, you can still see the processes created by lab2.

How many "zombies" appear on Terminal B?

There are 2 zombies that appear on Terminal B.



Google "zombie Unix processes". **Based on your experience** in this hands-on laboratory exercise, what is a "Zombie" process?

A zombie is also a defunct process but is a process that has execution completed but still shows in the process table.

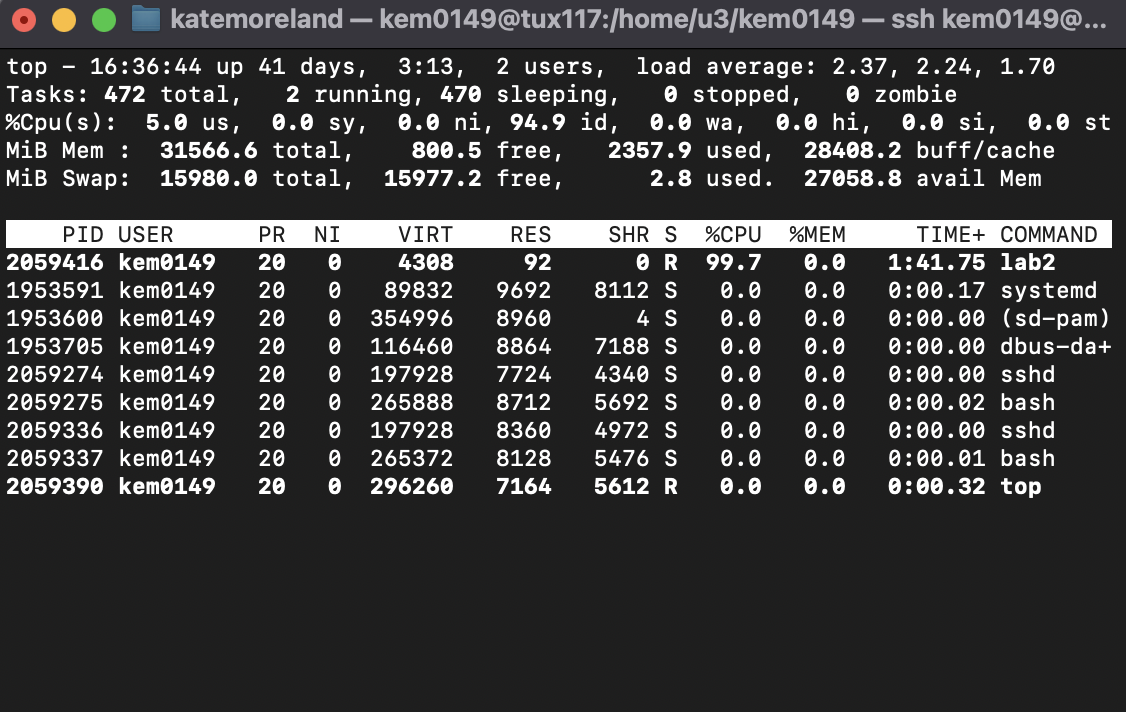
8) (7 points) On Terminal A, kill the parent process.

Do you see on Terminal B the processes created by the program **lab2**?

Yes the process created by lab2 still shows the child of the process.

How many "zombies" appear on Terminal B?

There are 0 zombies that show on Terminal B.

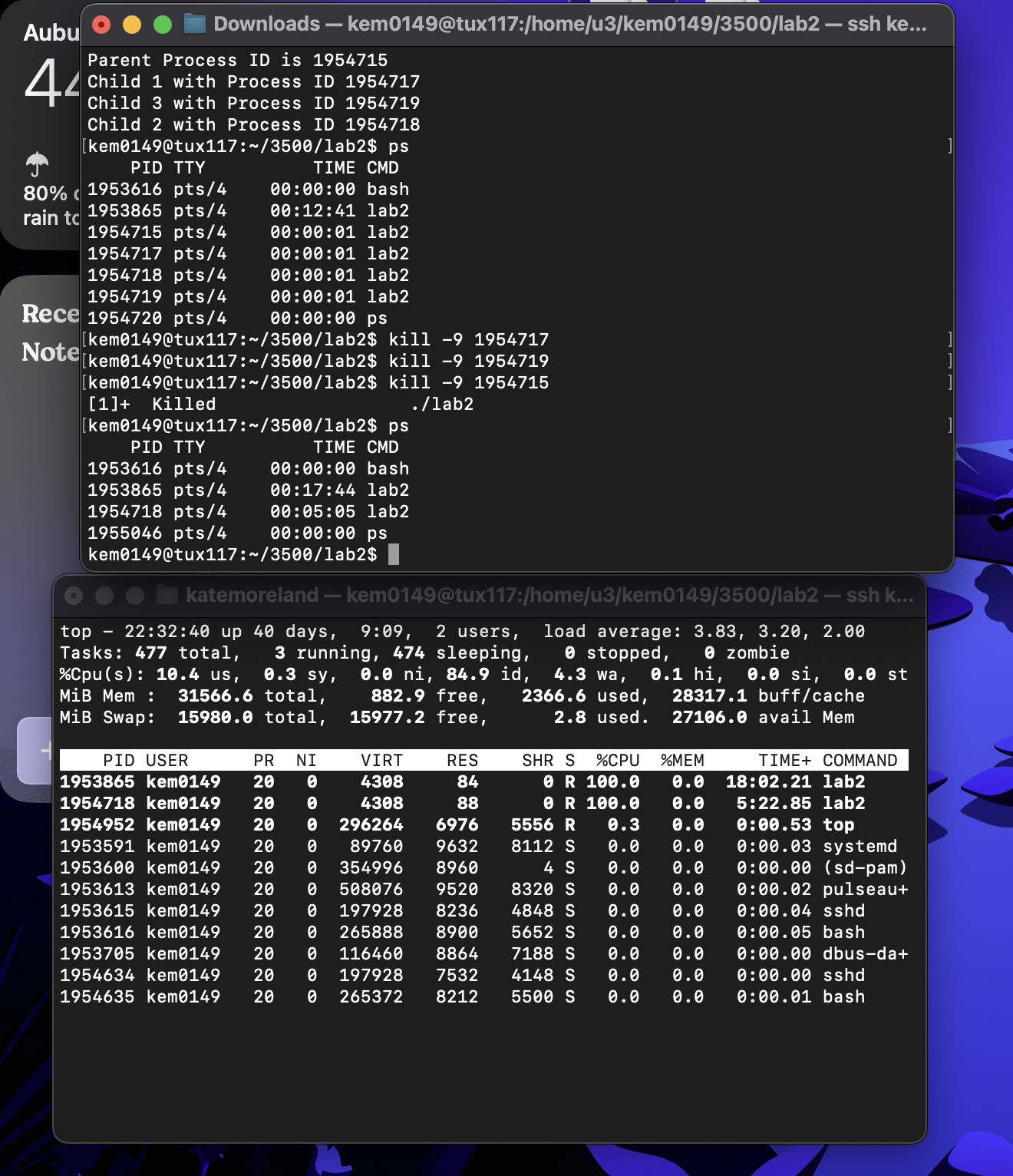


.

9) (7 points) Type on Terminal A the command ps

Does the information provided by **ps** on Terminal A match the information provided by top on Terminal B

Yes, the processes that show in Terminal A match the active processes shown in Terminal B.



Based on your experience with **ps** and **top**, what is (are) the key difference(s) between them?

Ps only shows a picture of the current processes at that moment and top updates in real time with the process.

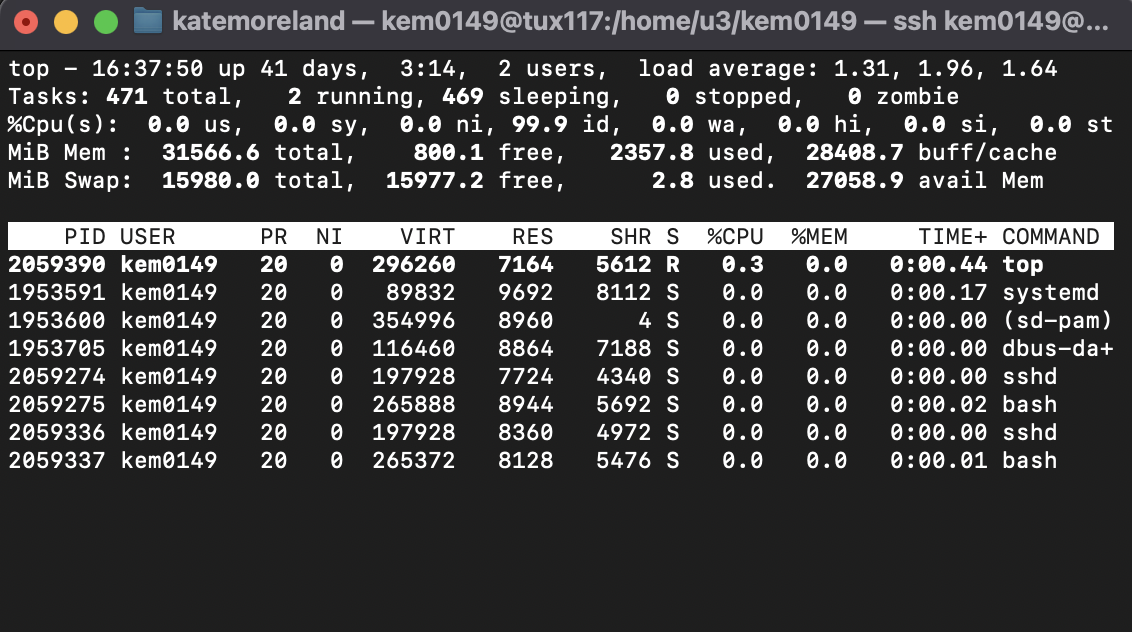
10) (6 points) On Terminal A, kill the last child process.

Do you see on Terminal B the processes created by the program lab2?

No, you do not see the process lab2 because it was killed.

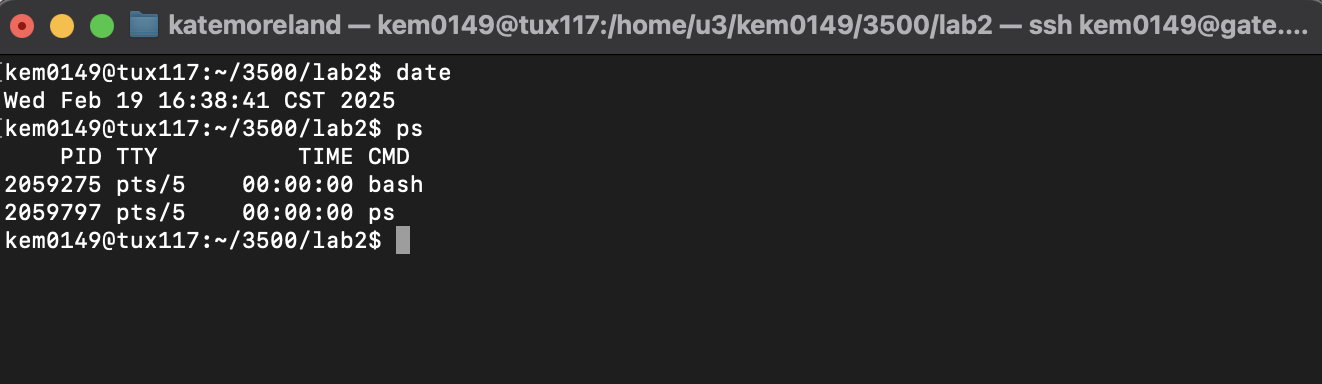
How many "zombies" appear on Terminal B?

There are 0 zombies on Terminal B.



Type **ps** on Terminal A.

It just shows the bash and ps.



Hit on Terminal B the key 'q' to leave the command **top**.

All the above questions are an **opportunity** to discover more about Unix OS commands. Use man and the Internet to learn more. Doing just the minimal will not make you proficient. **Knowing well Unix may help you in an interview for your dream job**.

Do not hesitate to ask questions or seek help on Piazza. Start early to avoid getting stuck at the last minute when your classmates and instructors are not available.

* What you need to turn in:
* Electronic copy of this file (including your answers) (standalone). Submit the file as a Microsoft Word or PDF file.
* Recall that answers must be well written, documented, justified, and presented to get full credit.
* How this assignment will be graded:
* A right answer will get full credit when:
* It is right (worth 25%)
* It is right AND neatly presented making it easy and pleasant to read. (worth 15%)
* There is an obvious and clear link between 1) the information provided in the exercise and in class and 2) the final answer. A clear link is built by properly writing, justifying, and documenting an answer (worth 60%).
* Calculation mistakes will be minimally penalized (2 to 5% of full credit) while errors on units will be more heavily penalized.
* You are welcome/encouraged to discuss exercises with other students or the instructor. But, ultimately, personal writing is expected.

**Appendix**: Grading: What is an OBVIOUS and CLEAR LINK?

Here is an example to explain what an **obvious and clear link** is and how we grade your work.

Consider the following problem:

"(100 points) John travels from Auburn to Atlanta in his car at a speed of 50 mph. Leaving at 8am, at what time will John reach Atlanta".

Here are the answers of three students and their scores:

**Student 1** answers: "10am". Student 1 will get 25 points.

**Student 2**answers : "John will reach Atlanta at 10am". Student 2 will get 25+15 = 40 points

**Student 3** answers: "The time t to travel a distance d at speed v is equal to d/v = d/50mph. The problem does not provide the distance d from Auburn to Atlanta. Based on Google, the distance from Auburn to Atlanta is approximately 100 miles (**document is here**). Therefore, the time t = 100 miles/50mph = 2 hours. Since John left at 8am, he will then reach Atlanta at 8am + 2 hours = 10 am".

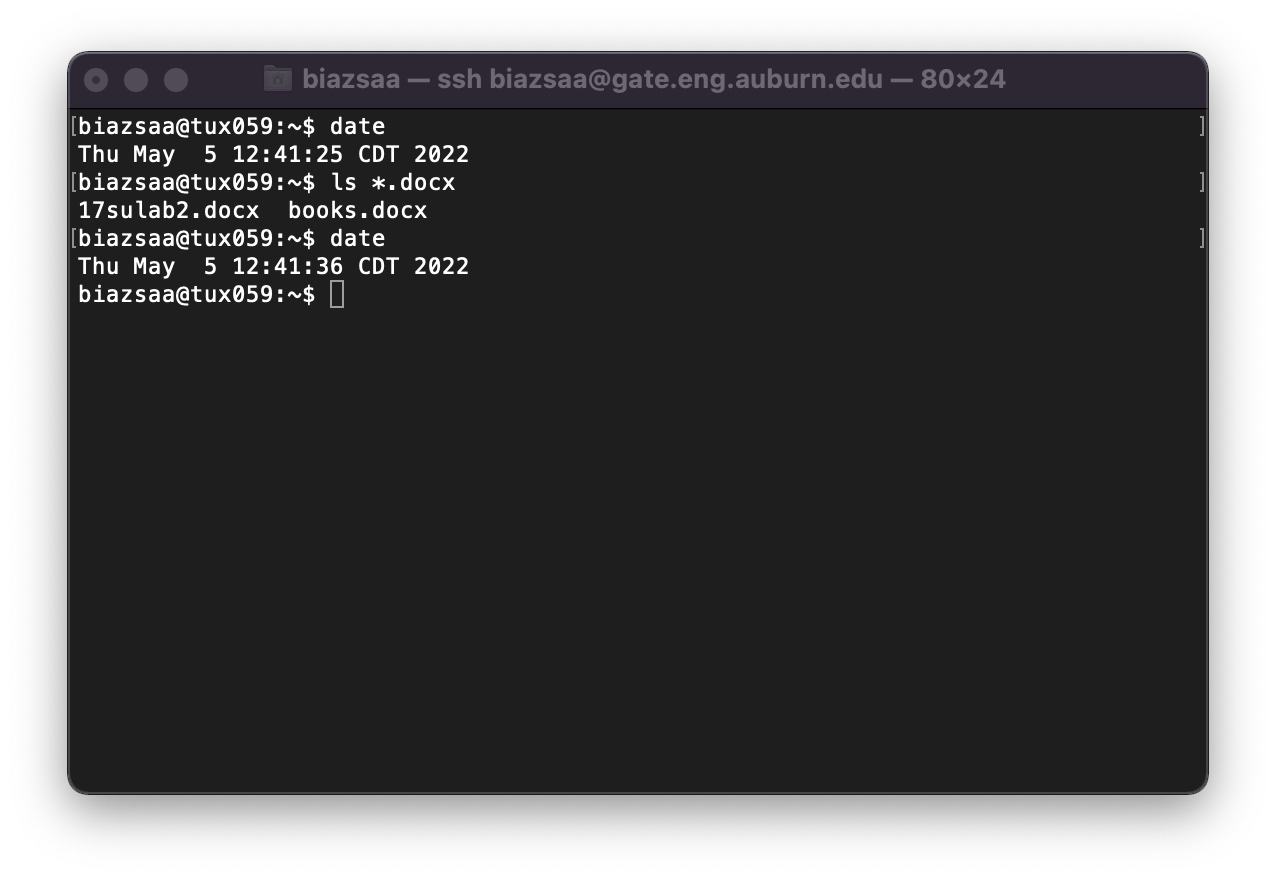
**Student 3** will get 25 + 15 + 60 = 100 points

Do you see the **direct** **link** going from the data provided in the question to the final answer, using general knowledge/formula and documents?.... Can you now solve the following problem and get 100 points?

"(100 points) Alice travels from Auburn to Atlanta in her car at a speed of 50 mph. Leaving at 8am, at what time will Alice reach Atlanta assuming that she had a flat tire that delayed her 30 minutes".

**Screenshot: Required Information**

**In order to save space, for this assignment and all *FUTURE* ones, clip out the screenshots to contain only the relevant information. *When applicable, ALL screenshots must show the date, the machine you are using for the exercise and the username of one of the team mates*. Make sure that the screenshots are easily readable. Below is template screenshot:**



1. See on the appendix what an obvious and clear link is. [↑](#footnote-ref-1)