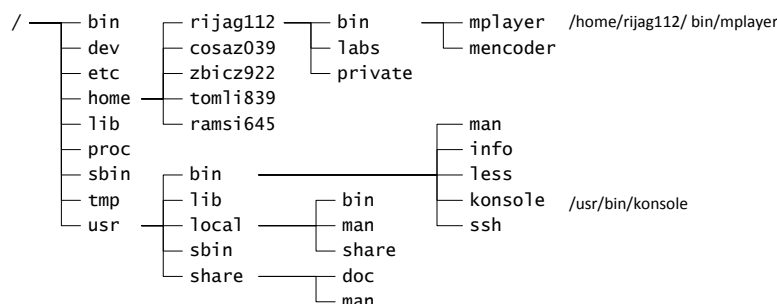


Lab 1: Basic Linux Familiarity Questions

Answer the following questions in your report. If you are instructed to carry out any activity, perform it and include your observation in your report. For this lab, you can use either `gedit` or `open office` to write the report. `Open office` might be preferable as you can insert images such as screenshot when required. Create a file in your computer, save it as `Lab1_reg1_reg2` (where `reg1` represents the registration number of one student and `reg2` represents the reg number of the second student) within the `Document` directory.

- Q 1. What is the purpose of tabs in a console? What happens when you open a tab and close a tab?
- Q 2. What is the purpose of the ***man*** command?
- Q 3. Which are the nine sections of the Unix manual?
- Q 4. Write down the meaning of f, K, m and D options of the ***man*** command?
- Q 5. Which section documents system calls?
- Q 6. Display the man page for the ***ls*** command.
 - a) When do you need to use the ***ls*** command?
 - b) What option to ***ls*** shows information about file sizes, owner, group, permissions and so forth?
 - c) What does the ***-R*** and ***-r*** options to ***ls*** do?

The figure below shows part of a Unix system.



- Q 7. In the example above name the relative and absolute path names indicating ***labs*** if
 - a) The current working directory is ***/usr/local/bin***
 - b) The current working directory is ***/home/cosaz039***
- Q 8. What are the differences between ***mv*** and ***cp*** commands?
- Q 9. It is possible to set individual permissions for user, group and others using ***chmod***. Review the documentation and answer the following questions:
 - a) How can you set the permission string to user read/write/execute, group write and others read using ***chmod*** in long and numeric formats?
 - b) How can you grant user and group write permissions without changing any other permissions in long and numeric formats?
- Q 10. What do the following numeric file modes represent:
 - a) 777
 - b) 634
 - c) 647
 - d) 467

- Q 11. How would you change the ownership of a file called `hello.txt` to `karim` under the group of `hexa`?
- Q 12. Write down the command to create a symbolic link called *pointer* for a file called *foobar* within your home directory.
- Q 13. Use the **env** command to display all environment variables. What is `PATH` set to (you might want to use **grep** to find it, also man it to understand the functionality of `grep`)? What is this variable used for (the man pages for your shell might be helpful in answering this question)?
- Q 14. How would you create an environment variable? How would you change the value of an existing environment variable?
- Q 15. Where will `stdout` and `stderr` be redirected for: `command 2>&1 > file1`?
- Q 16. Create a long running process in the background by typing `ping 127. 0. 0. 1 >/ dev/ null &`. Describe how to find its process id and kill it.
- Q 17. When would you use **`kill -9 pid_number`** command? What is its difference with **`kill -3 pid_number`**?

Carry out the following activities.

- Create a file called **`test`** in command prompt in the Documents directory within home directory
 - Use the *nano* editor to open the file and explore the file.
 - Add a few lines of text in the file.
 - Cut a line of texts in the file.
 - Justify a line of texts.
 - Save the file and then exit.
- Q 18. What are the commands to cut and justify texts in *nano*?
- Q 19. Which commands you have used to save and exit from the file?
- Q 20. Write a shell script that will take three numbers and display the result of their addition, subtraction, multiplication and division using different functions in the console.
- Q 21. **[BONUS]** Write a shell script that tests if two strings and numbers are equal or not by taking inputs from the console and then displaying out the appropriate message.
- Q 22. **[BONUS]** Write a shell script that takes two string inputs and then outputs them into another file called **`output`** along with the home directory of the user and present working directory (which you can get using the **`pwd`** command).