**Part I: Questions**

1. **What is a Graphical User Interface (GUI) and how does that differ from the Command Line Interface (CLI)?**

In general, GUI is designed to be easier to use using graphical elements and mostly can be used by a click, however CLI allow to interact with our system using command lines.

1. **What does the shell do?**

Shell is a command line interface which allow us to control and do several things in our computer using command line. Example can be using command lines for moving files, copying files, etc.

1. **What is the command used for listing things in a directory?** ls
2. **What is the command used for changing directories?** cd
3. **What command would you use to get your current working directory?** pwd
4. **How do you get the manual for these commands?** man (name of command line)
5. **What does the shell prompt look like?** $ sign
6. **How would you list things in a directory in chronological order?** Ls -t
7. **Name two ways to get to the ‘home’ or ‘root’ directory?** cd / or cd ~
8. **What is the difference between an absolute path and a relative path?**

In general, I can tell that the absolute path shows the locations of our file from root directory, but relative path just show the current directory and not root.

1. **What are the two relative path directories we talked about and what do they mean?**

‘..’ show directory to the parent (../Desktop), “~” own home directory

1. **Why shouldn’t you put spaces in filenames?**

In Linux and other programming languages, when we create a file, vector or object they need proper language, otherwise program cannot identify the file name. It is common to use dash line.

1. **Name a way to have multiple readable words in a filename without spaces?**

We can use dash line or connect the words together. Also, we can use words with space when you put them in quotes

1. **What is the program nano, what does it do? Do you use nano? If not what do you use?**

I don’t use nano, use Jupiter notebook instead.

But nano in general is a text editor particularly for Linux which can help us to edit and work on text.

1. **\* is a form of a wildcard. What does it mean?**

It means all the list of words or numbers, as an example (\*.jpg mean all the jpg files in the directory).

1. **The following code would match what? ls \*.txt**

It means, show us the list of all the files which have txt format.

1. **Why do we need to be careful with the mv command?**

It is possible to move the files to the new directories and remove the file from origin. Also, it can rename the names.

1. **Why do we need to be careful with the rm command?**

It is possible to remove all the file permanently from your directory. Also, it can damage our system

1. **What is the difference between > and >>?**

“>” is a redirection operator, and “>>” also is an output operator but it appends all the files which already exist in datafile

1. **What does head do?**

It shows number of top rows in a file. For example, head (5) show top five rows

1. **What is the purpose of | (pipe)?** pipe use to connect two or more command line, also we can send output of one command to another one
2. **Interpret the following command: cat huge\_file.fasta | uniq | head -n 5 >proteins.fasta**

Read the huge\_file.fasta file, select the uniq files and put the top rows in the new file ans choose the name as proteins.fasta

1. **What is a loop and when would you use it?**

Loop repeating a repetitive part of code, and it is great for saving time, and increase the speed of a model.

1. **In a directory with the following files. What would the following loop do?**

**$ for filename in \*.txt; do**

**> cat ${filename} >>fruits.txt**

**> done**

It is a loop, and it means go through all the files that have txt format in the directory, and concatenate the files in a new file as fruits.txt

1. **What is a shell script?**

It is a command line approach which use in terminal. It makes our job easier (for example in case of finding a file, moving, editing, and changing the names) so in this case we don’t need to do manual work.

1. **What are the benefits of writing and running a script over typing the code in?**

Later it is easier to use and is more repeatable

1. **When looking at a script what does # mean? Why would you use one?**

#In our case, it means description of a code. Normally, it helps our code to be more producible and easier to use by yourself and others, especially when move it to the GitHub.

1. **What does grep stand for and what does it do?**

Grep stand for Global regular expression print, and it helps to filter the searches to find a particular pattern in our file and show the lines which show these patterns.

1. **What does find do?**

It helps us to find a directory and locations of the files that we need.

## Part II - Writing Code

As part of this exercise please open the shell and practice moving around into different files. Best practice is to try a little bit every day. Try to challenge yourself by not using the mouse. For each of the questions below type your code and the result from the shell prompt.

1. **Open the shell and change directories to the desktop and list the contents, paste your code here.**

/Users/Masoud # main directory

masoud@Masouds-MacBook-Pro ~ % cd Desktop # change directory ro the Desktop

masoud@Masouds-MacBook-Pro Desktop % pwd # check the directory

/Users/masoud/Desktop # now our directory is Desktop

1. **Draw the file structure from the root directory into one of the folders on the desktop. You can submit a pdf with a drawing. Select three locations and type the commands for changing directories to those locations use a combination of absolute and relative paths.**

Attached to the email

1. **On your computer create a directory on your desktop. Create two files without opening them that have file extensions .txt. Then use wildcards to list the files in that directory. Paste you code here.**

Cd ~

Cd Desktop

touch train.txt

touch test.txt

ls \*.txt

(test.txt, train.txt)

1. **Download the file [Hutia\_DNA.fasta](https://www.dropbox.com/s/07y610c8zt63nt5/Hutia_DNA.fasta?dl=0) from the Github repo. Make a new directory on the desktop. Move the file into that directory. List the contents of the directory. How many lines are in this file? Fasta files are a type of data file that holds DNA sequences. They are all formatted the same. The first line starts with a > and a name followed by a line break, and then DNA sequence followed by a line break. The next line starts with an > and another name followed by a line break and the DNA sequence. With that in mind, how could you use linux commands to determine how many sequences are in this file? How many sequences are in this file?**

I could not find the [Hutia\_DNA.fasta](https://www.dropbox.com/s/07y610c8zt63nt5/Hutia_DNA.fasta?dl=0) in the github, so I just choosed one the text files and write the command line,

Cd ~

Cd Desktop

masoud@Masouds-MacBook-Pro Desktop % mkdir DNA

masoud@Masouds-MacBook-Pro Desktop % mv amino-acids.txt DNA

masoud@Masouds-MacBook-Pro Desktop % pwd

/Users/masoud/Desktop

masoud@Masouds-MacBook-Pro Desktop % cd DNA

masoud@Masouds-MacBook-Pro DNA % pwd

/Users/masoud/Desktop/DNA

masoud@Masouds-MacBook-Pro DNA % ls -l

total 8

-rw-rw-r--@ 1 masoud staff 283 Jul 30 01:41 amino-acids.txt

masoud@Masouds-MacBook-Pro DNA % wc -l amino-acids.txt

20 amino-acids.txt

masoud@Masouds-MacBook-Pro DNA %

masoud@Masouds-MacBook-Pro DNA % wc -w amino-acids.txt

42 amino-acids.txt

masoud@Masouds-MacBook-Pro DNA %

1. **Build a single line of code that would take the first 100 sequences of this file and put them in a new file called ‘MyOutputFile.fasta’. Use a combination of wc, sort and uniq in a single line of code to tell me how many unique lines are in this file. Paste your commands and the answer here.**

wc -l amino-acids.txt | split -l 10 amino-acids.txt | sort | uniq -c

1. **Write code that would create a loop to copy all the files in one directory ending in .txt to another directory. Paste the code here.**

for file in $(ls -a $1)

do

cp $1/$ amino-acids.txt Desktop

done

1. **Write a bash script with in-line documentation (hint #) to show how to run a made up python program (script) called 'find\_taxa.py' on set of files ending in ‘.fasta’. Show in the script how to run the program on each of those files. What is the name of your script? Type the script here.**

I did not understand the questions, I will ask you on Thursday questions.

1. **From the Hutia\_DNA.fasta file tell me how on how many lines do we find the pattern ‘GAGA’. What was the code used to find this?**

grep -c Glut amino-acids.txt (Find number of Glut pattern in the amino-acids.txt).

**#!/usr/bin/sh**