Homework 1

Part I:

1. GUIs are widely used and rely on click on instructions and buttons to perform a task. Generally, these are more user friendly. On the other hand, CLIs depend on writing out the commands but are much more versatile and powerful. More useful when repetitive tasks need to be executed more efficiently.
2. Shells are programs that run other programs.
3. ls
4. cd
5. pwd
6. man
7. $
8. ls -t
9. Using absolute or relative paths
10. An absolute path lists the full path starting from the root followed by all directories on the way to the location or file, while the relative path is defined in relation to the current directory the user is in.
11. One is the current directory “.”. The second one is the parent directory “..”.
12. Spaces between words result in the name being read as two different commands.
13. Using ‘\_’.
14. Nano is a text editor; it is used to write scripts that can be saved and executed later. I use nano.
15. It can represent any character or string of characters.
16. It would list all the text files in the current directory.
17. This is the move command that deletes files from the original location and can re-name them in the new location, so it is important to be careful with it and keep track of file names and locations.
18. Remove command; will delete files.
19. The >> appends to a file or creates the file if it doesn't exist. The > overwrites the file if it exists or creates it if it doesn't exist.
20. Head lists the first n (10 by default) lines of the file.
21. It lets you sends the output of one command to another; redirects the command.
22. This would read the file huge\_file.fasta, then eliminates repeated lines, then writes the first 5 lines of the huge\_file.fasta (without the repeated lines) into a file proteins.fasta.
23. Loops repeat the same command or set of commands for x number of times. It is used for repetitive actions.
24. For each text file in the directory, read the content and write the content in the file fruits.txt (appending the file each time).
25. Shells script is a file containing a list of commands and it is used to automate a set of instructions to be executed one after the other, instead of typing in the commands one after the other multiple times.
26. Saves time and prevents errors. Set of instructions are automated to be executed one after the other.
27. # indicates part of the script that does not run. It is used for annotating the script.
28. It is used to search for a string of characters in a particular file.
29. Find locates files.

Part II:

1. cd Desktop

ls

A screenshot of a phone

Description automatically generated with medium confidence

To move to folder “POLS\_manuscript” from home directory “Desktop”:

cd KARLA/POLS\_manuscript

To move to “Desktop” from home directory “KARLA”:

cd ..

To move to “kalujevic” from “KARLA”

cd ../..

1. mkdir Folder\_1

cd Folder\_1

touch file1.txt

touch file2.txt

ls \*.txt

1. cd ..

mkdir Folder\_2

cd ../downloads

mv Hutia\_DNA.fasta ../Desktop/Folder\_2

cd ../Desktop/Folder\_2

ls

wc -l Hutia\_DNA.fasta

# 2382004 lines

grep ‘^>’ Hutia\_DNA.fasta | wc -l

# 1191002 sequences

1. If we want the full sequence including the name:

head -n 200 Hutia\_DNA.fasta > MyOutputFile.fasta

If we only want the sequence (only second lines):

grep ‘^[^>]’ Hutia\_DNA.fasta | head -n 100 > MyOutputFile.fasta

1. Script1.sh

#!/usr/bin/sh

for file in \*.txt

do cp $file ../Newfolder

done

sh Script1.sh

1. ###############

## find\_taxa.py ##

###############

#!/usr/bin/py

for file in \*.fasta # for all fasta files in the directory

do cp $file ../Newfolder2 # copy them into a new directory

done

1. grep ‘GAGA’ Hutia\_DNA.fasta | wc -l

#304806

Or

grep -c ‘GAGA’ Hutia\_DNA.fasta