Basic Unix Commands:

Open Unix terminal and type

date

echo hello

**Content Representation:**

Files, Directories, Directory structure, Paths

**/**  Every level of hierarchy is marked by a slash

**cd** To change directory we write cd and then full path or relative path to where we want to go

**cd .** To move to the current directory

**cd ..** To move to the parent directory

**pwd** This prints current working directory or current location within the file system

**ls** This gives us a list of files under the current directory in alphabetical order and filenames with uppercase letters are listed first.

**ls -l** This gives additional information about the files-whether it is directory d or file r, permissions, the number of links to the file eg 1, owner name for the file, size of the file in bytes, date when it was last modified, file name

**ls -lt** To see the files in the order in which they were created or reverse chronological order

man followed by command eg man ls To see options for the commands anytime

**File Naming:**

**Wild Card:**

**\***  The star represents any number of letters or numbers.

**ls filename\*** So we can list files of same name but different types by using \* . So ls HW2\* will list all files that have the name home\_work eg home\_work.genome, home\_work.samples etc

**?** It stands for just one character.

**ls ?ome\_work.genome** will return home\_work.genome

**[ ]** To represent a range of characters

**ls [a-z]\*.genome** gives all files with the extension genome

Combinations

**ls p\*.genome** Lists all the files that start with letter p and have extension genome

**{ }**

**ls {home\_work,quiz}.genome** This will list all the files that have either of the names with extension genome

**Content Creation and Removal:**

**mkdir directoryname** To create the directory eg. mkdir Home\_Work

**cp** To copy single or multiple files to a given directory

**cp home\_work1 Home\_Work** to copy file home\_work to directory Home\_Work

**cp home\_work1, hoem\_work2, home\_work3 Home\_Work** to copy multiple files or

**cp home\* Home\_Work**

**mv** To move files into directories

**home\_work1.ipynb Home\_Work** moves file into directory

Note: If directory does not exist then it will rename the file to that name

**mv home\_work1** Home\_Work will rename the file home\_work1 to Home\_Work if the directory does not exist

**mv home\_work\*** Home\_Work will move several files with the name home\_work to the Home\_Work directory.

**Remove files**

**rm home\_work1** This will remove the file from the current directory. We will get a prompt is we want to actually remove the file and if we type y the file will be removed and if we type n the file will not be removed. The action is not reversible.

Remove directories

**rmdir directoryname** To Remove the entire directory we need to delete all its contents files and subdirectories then

**rmdir Home\_Work** removes the directory

OR

**rm -r** To remove the directory with all its contents recursively. Type y at all prompts

**Accessing Content:**

**more filename**  To see contents of file, Spaccebar to move down file

**less filename**  To see contents of file. UP and DOWN Arrow keys to move up and down file. Q to get out of file

**/ >** To see the next sequence in a fasta file

**head filename** To see top 10 lines.

**head -5** To see top 5 lines

**tail** To see last 10 lines

**tail -5** to see 5 lines

cat To concatenate the contents of several files

**cat home\_work\*** To concatenate contents of all files starting with home\_work

**Standard Input & Standard Output:**

**cat** To display content of file to terminal

**wc** To get line, word, character counts printed to terminal

**wc filename**

78 82 2085 phix.fa

**wc -l filename** To print just number of lines in file

78

**>** To redirect output to a file

**-wc -l phix.fa > nlines** Will store output, number of lines in file phix.fa, to a new file named nlines

**more nlines**

78

**<** To get output from file and display on terminal

**wc -l < phix.fa**

78

**Piping |** To pass output of one command to the next

**ls | wc -l**

Will get the output of ls and give it to the wc -l so we get the number of files in the current directory

**cat home\_work\* | less** Will concatenate all files with name home\_work and display on terminal

**Querying Content:**

**sort filename** To sort contents of the file in alphabetical order by default

**sort -r filename** Will sort contents of file in reverse alphabetical order

**Sort single column**

**sort -k 2 filename** Will sort the file by alphabetical order of col 2

**sort -k 2n filename** Will sort the file by numerical order of col 2

**sort -k 2nr filename** To sort column 2 in reverse order

**Sort multiple columns**

**sort -k 3 -k 2n filename** To sort by col 3 in alphabetical order and then by col 2 in numerical order

**sort -k 3 -k 2nr filename** To sort by col3 in alph order and col2 in reverse numerical order

**Individual fields within a file**

**Tab delimited files**

**cut** delimits the file into fields at the tab

**cut -f1 months.txt** Will get the first column of the file months

**cut -f1, 2 months.txt** Will get the first and 2nd column of the file months

**cut -f1-3 months.txt** Will get the first to 3rd column of the file months

**Delimit by character**

**cut -d ‘ ‘ -f1 months.txt** To cut by space and make get the first column( will make a column when it encounters a space “ ‘ and it will get the first column -f1from the file

**cut -d ‘ ‘ -f1,2 months.txt** To cut by space and get 1 and 2 cols

**cut -d ‘ ‘ -f1-3 months.txt** To cut by space and get 1-3 cols

Cut from file and paste in new file

**cut -d ‘ ‘ -f3 months.txt > seasons** Cuts col 3 from file and puts in new file seasons

Unique objects in file

**sort -u seasons** Sorts the seasons file in alphabetical order and returns sorted unique objects.

**uniq** Looks at file and analyzes all lines and we will get all unique elements in a certain group and some objects may occur more than once if they appear in different locations of the file

**uniq | sort -u** Will take the input from the uniq and get unique objects such that they appear only once

**uniq -c** To see how many times a particular word occurs in a particular context

**grep fall \*.txt** This will get us all txt files in current directory which have the word fall

**grep ‘ 12 winter’ \*.txt** Will get us the line in our file where the text within the quotes is found

**grep -n ‘ 12 winter’ \*.txt** Will give us the number of the line where the expression is found

**grep -v “ 12 “ filename** This will ignore the character and get all lines that begin with something other than what is in the bracket

**Comparing Content**

**diff file1 file2** Compares files line by line and returns the minimum number of changes we need to make to make one file into another

Uses the sequence alignment algorithms

**comm file1 file2** Compares two files line by line and if sorted then and col1 of output tells us the lines unique to file1 col2 lines unique to file 2 and col3 lines common to both

comm -1 - 2 file1 file2 will give will ignore line unique to file1 and file2 and return the lines common to both files

comm -3 will return 1st and 2nd cols ie line unique to file1 and file2 respectively and ignore col3 ie lines common to both

This can be done in any combination comm -1 -3, comm -2 -3 etc and with other commands using pipe operator comm -1 -2 file1 file2 | wc -l to get number of lines

**Archiving Content**

**gzip** filename To compress one file at a time, Uses lempel ziv algorithm

**gunzip** filename To uncompress file

**bzip2** filename To compress single file slightly more than gzip Uses Burrow wheeler algorithm

**bunzip2** filename To uncompress file

tar -cvf Archivefilename file1 file2 file3 This takes all the files and stores them in target archive file without compressing them

tar -cvf Virus.tar lambda phix will give us

Virus.tar file which contains lambda and phix files

**Common Genomics Examples:**

**How many chromosomes in the genome?**

**grep “>” filename** To get the list of chromosomes in a file

**grep -c “>” filename** To directly get count from grep command OR

**grep “>” filename | wc -l** To count the number of chromosomes in a file

**How many genes and variants in the genome?**

**cut -f1 filename | sort -u | wc -l** To get number of genes or transcripts in the genome first cut col then sort and then count

**cut -f1 filename | uniq | wc -l** If the file is sorted then cut the column number of interest here 1 and then list unique values and then count them to get the number of genes OR

**How many genes have 1 variant**

**cut -f1 filename | uniq -c | grep “ 1 “ | wc -l** This will get the col and then get the count for the number of unique values and then it will get the values that have count 1 and count the number of those values

**How many genes have more than 1 variant**

**cut -f1 filename | uniq -c | grep -v “ 1 “ | wc -l** This will get the col and then get the count for the number of unique values and then it will get the values do not have count 1 and count the number of those values