EXERCISE 1

Question 1

1. ls /usr/bin
2. man ls
3. ls /usr/bin -l
4. ls /usr/bin/a*
5. . is current directory which is the bin
   .. is the parent directory which is the usr
6. ls /usr/bin/*.py

Question 2

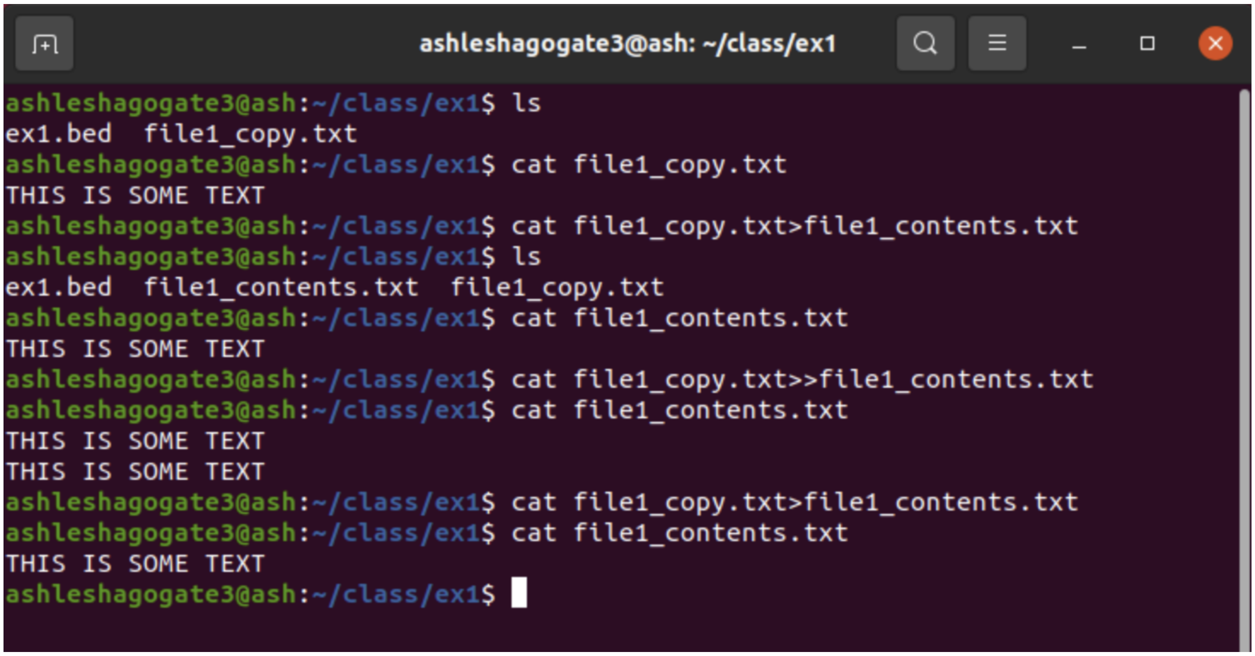
1. vi file1.txt
2. Press i to enter insert mode and then type out your text "this is some text"
   i
   This is some text
3. Press esc key on the keyboard to get out of insert mode.
   :wq
   Press return on keyboard. The file is saved and we are back to our file structure.

Question 3

1. cp file1.txt file1_copy.txt
2. rm file1.txt
3. ls

Question 4

1. cat file1_copy.txt
2. cat file1_copy.txt>file1_contents.txt
3. cat file1_contents.txt
4. cat file1_copy.txt>>file1_contents.txt
5. cat file1_contents.txt
6. cat file1_copy.txt>file1_contents.txt
   cat file1_contents.txt
   The > command does not append the existing text in file1_contents.txt, unlike the >> command. Instead, it starts fresh. The outputs are shown in the screenshot below.



```
ashleshagogate3@ash:~/class/ex1$ ls
ex1.bed  file1_copy.txt
ashleshagogate3@ash:~/class/ex1$ cat file1_copy.txt
THIS IS SOME TEXT
ashleshagogate3@ash:~/class/ex1$ cat file1_copy.txt>file1_contents.txt
ashleshagogate3@ash:~/class/ex1$ ls
ex1.bed  file1_contents.txt  file1_copy.txt
ashleshagogate3@ash:~/class/ex1$ cat file1_contents.txt
THIS IS SOME TEXT
ashleshagogate3@ash:~/class/ex1$ cat file1_copy.txt>>file1_contents.txt
ashleshagogate3@ash:~/class/ex1$ cat file1_contents.txt
THIS IS SOME TEXT
THIS IS SOME TEXT
ashleshagogate3@ash:~/class/ex1$ cat file1_copy.txt>file1_contents.txt
ashleshagogate3@ash:~/class/ex1$ cat file1_contents.txt
THIS IS SOME TEXT
ashleshagogate3@ash:~/class/ex1$ 
```

Question 5-

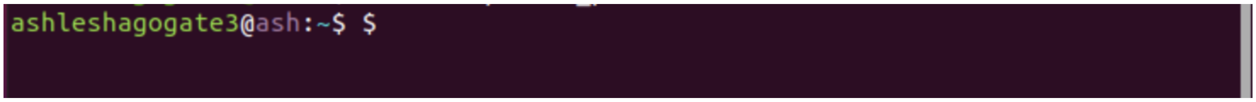
1. cd ~/class/ex1
2. head ex1.bed
   tail ex1.bed
3. head -50 ex1.bed
4. tail -25 ex1.bed
5. more ex1.bed
6. less ex1.bed

Question 6

1. mkdir myDir
2. rmdir myDir
3. mkdir myDir
4. cp file1_contents.txt myDir
5. Failed to remove. Directory not empty
6. Cannot remove as it is a directory
7. mkdir -p dir1/dir2

Question 7’

1. echo $PS1
2. ls -a
3. vi .bash_profile
4. Pasted the code
5. done
6. See screenshot.



```
ashleshagogate3@ash:~$ $
```

The colours did not change automatically when I started a new terminal. I had to run source ~/.bash_profile
I saw this solution on stack overflow. Another recommendation was to edit the bashrc but I was hesitant to do that.

Question 8

1. cd class/ex1
2. wc -m ex1.bed
   Ans- 78290
3. wc -l ex1.bed
   Ans- 3414

Question 9

1. perl -e 'foreach(1..100){print $_."\n"; print STDERR ($_ / 2)."\n"}'
2. perl -e 'foreach(1..100){print $_."\n"; print STDERR ($_ / 2)."\n"}' >myOut.txt
3. perl -e 'foreach(1..100){print $_."\n"; print STDERR ($_ / 2)."\n"}' 2>myErr.txt
4. perl -e 'foreach(1..100){print $_."\n"; print STDERR ($_ / 2)."\n"}' >myOut.txt 2>myErr.txt
5. perl -e 'foreach(1..100){print $_."\n"; print STDERR ($_ / 2)."\n"}' >mySeq.txt 2>&1

Question 10

1. seq 0 .5 100 > longSeq.txt
2. cat longSeq.txt | head -50 | tail -1
3. cat longSeq.txt | tail -n +14
4. cat longSeq.txt | head -n -13
5. cat longSeq.txt |head -50| tail -6|wc -m
   Ans- 30