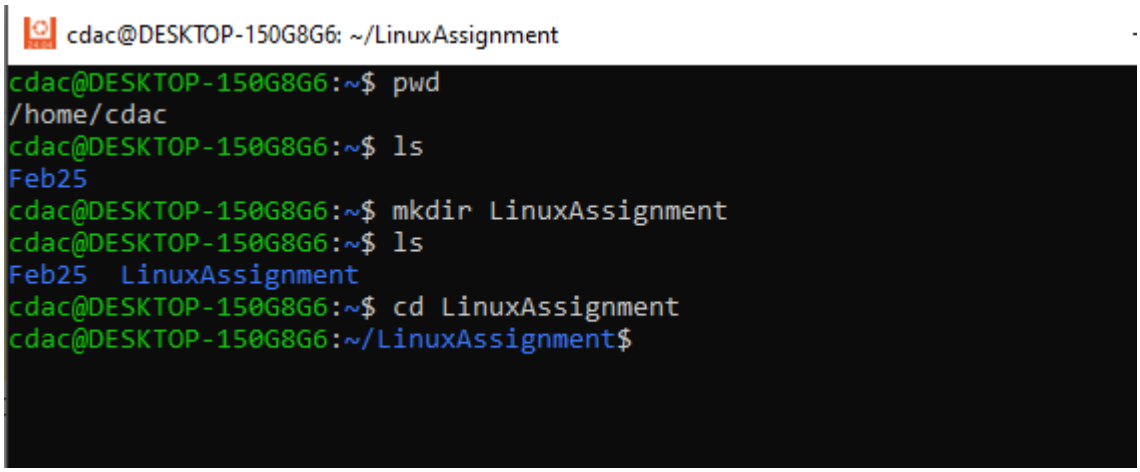
Concepts of Operating System

Assignment 1

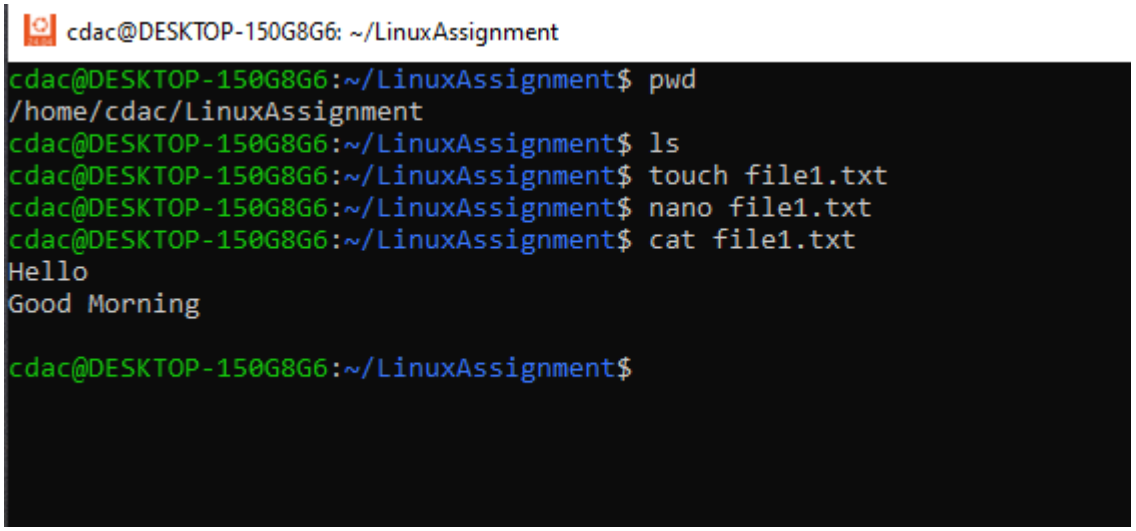
Name - Sanika Karade

Problem 1-

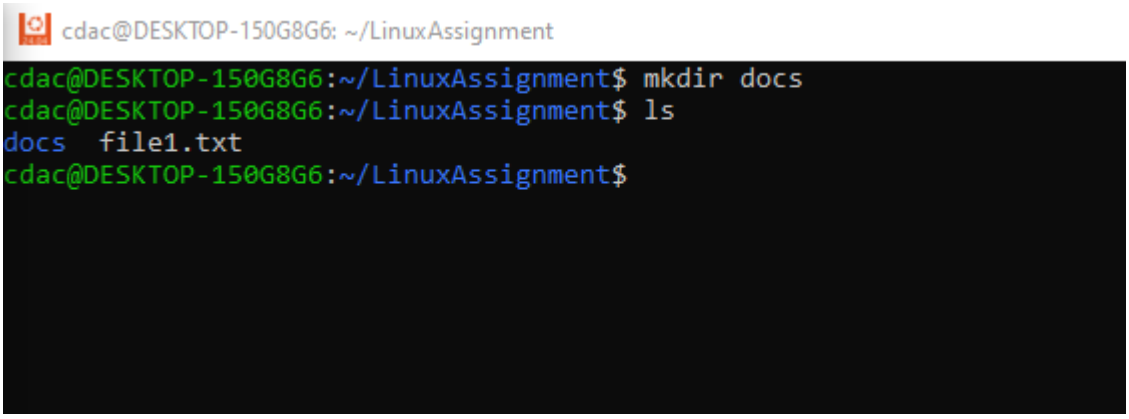
a) Navigate and List: a. Start by navigating to your home directory and list its contents. Then, move into a directory named "LinuxAssignment" if it exists; otherwise, create it.



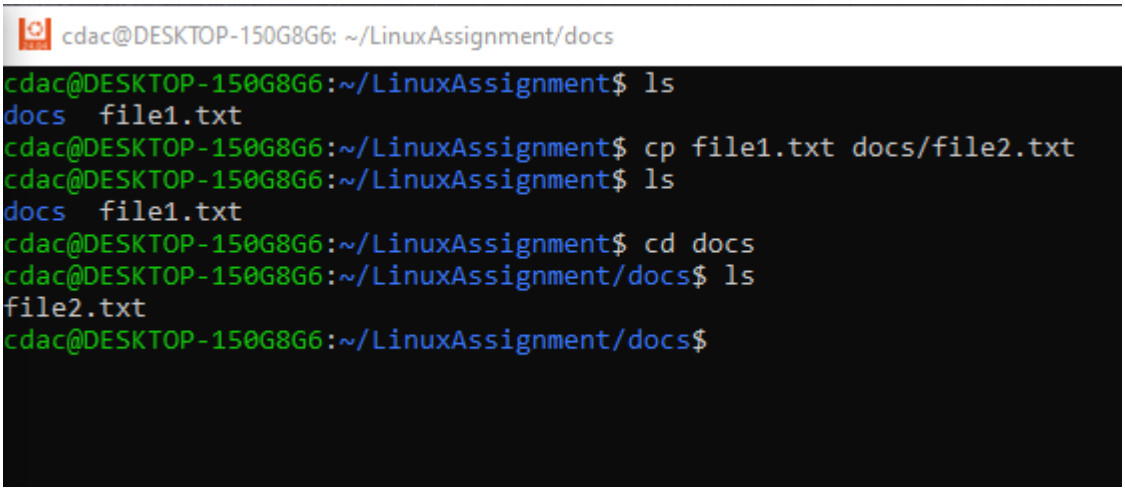
b) File Management: a. Inside the "LinuxAssignment" directory, create a new file named "file1.txt". Display its contents.



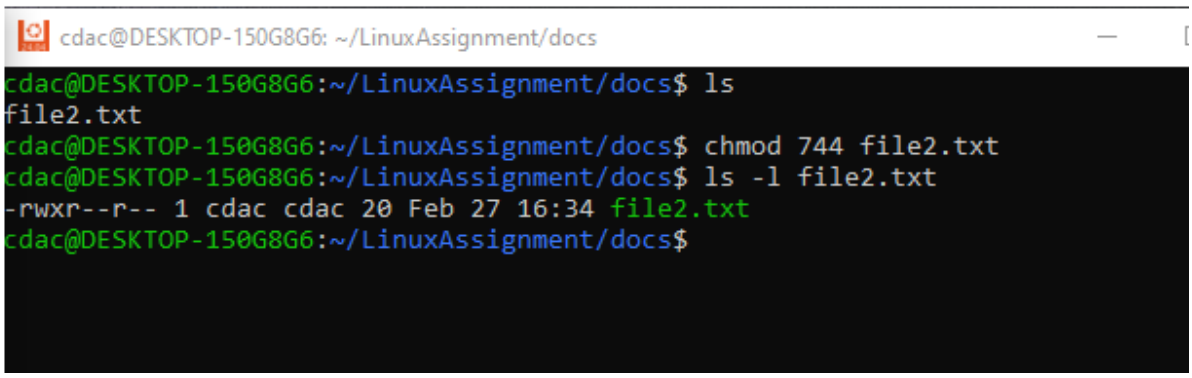
c) Directory Management: a. Create a new directory named "docs" inside the "LinuxAssignment" directory.



d) Copy and Move Files: a. Copy the "file1.txt" file into the "docs" directory and rename it to "file2.txt".



e) Permissions and Ownership: a. Change the permissions of "file2.txt" to allow read, write, and execute permissions for the owner and only read permissions for others. Then, change the owner of "file2.txt" to the current user.



```
cdac@DESKTOP-150G8G6: ~/LinuxAssignment/docs
cdac@DESKTOP-150G8G6:~/LinuxAssignment/docs$ chown cdac:cdac file2.txt
cdac@DESKTOP-150G8G6:~/LinuxAssignment/docs$ ls -l file2.txt
-rwxr--r-- 1 cdac cdac 20 Feb 27 16:34 file2.txt
cdac@DESKTOP-150G8G6:~/LinuxAssignment/docs$
```

f) Final Checklist: a. Finally, list the contents of the "LinuxAssignment" directory and the root directory to ensure that all operations were performed correctly.

```
cdac@DESKTOP-150G8G6: ~/LinuxAssignment
cdac@DESKTOP-150G8G6:~/LinuxAssignment$ ls
docs  file1.txt
cdac@DESKTOP-150G8G6:~/LinuxAssignment$ ls /
bin                etc   lib.usr-is-merged  opt   sbin                sys
bin.usr-is-merged  home  lib64              proc  sbin.usr-is-merged  tmp
boot               init  media              root  snap                usr
dev                lib   mnt                run   srv                 var
cdac@DESKTOP-150G8G6:~/LinuxAssignment$
```

g) File Searching: a. Search for all files with the extension ".txt" in the current directory and its subdirectories.

```
cdac@DESKTOP-150G8G6: ~
cdac@DESKTOP-150G8G6:~$ find . -name "*.txt"
./LinuxAssignment/docs/file2.txt
./LinuxAssignment/file1.txt
cdac@DESKTOP-150G8G6:~$
```

b. Display lines containing a specific word in a file (provide a file name and the specific word to search).

```
cdac@DESKTOP-150G8G6: ~/LinuxAssignment
cdac@DESKTOP-150G8G6:~/LinuxAssignment$ grep "Good" file1.txt
Good Morning
cdac@DESKTOP-150G8G6:~/LinuxAssignment$
```

h) System Information: a. Display the current system date and time.

```
cdac@DESKTOP-150G8G6: ~
cdac@DESKTOP-150G8G6:~$ date
Thu Feb 27 16:50:50 UTC 2025
cdac@DESKTOP-150G8G6:~$
```

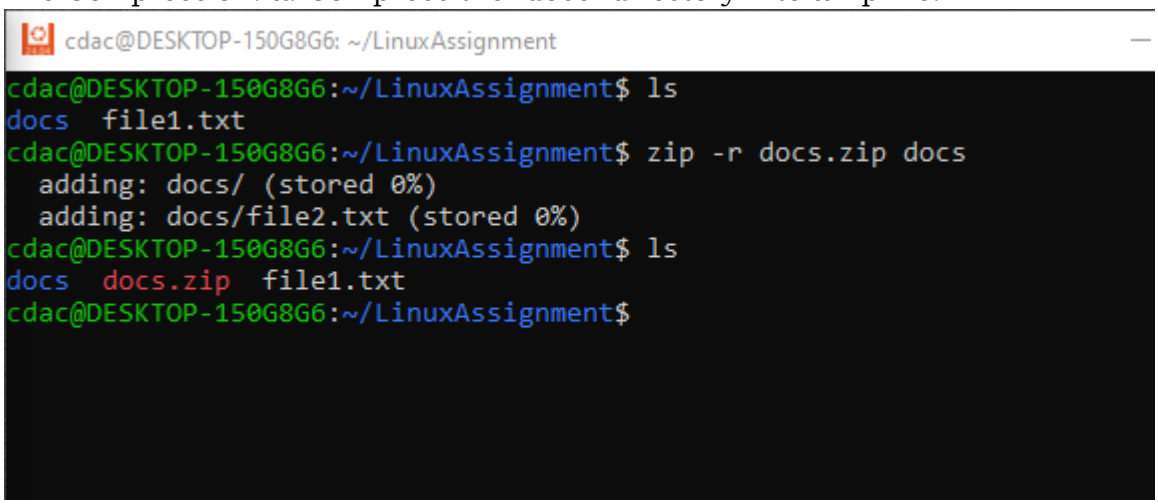
i) Networking: a. Display the IP address of the system.

```
cdac@DESKTOP-150G8G6: ~
cdac@DESKTOP-150G8G6:~$ ifconfig
lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 1500
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0xfe<compat,link,site,host>
        loop  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wifi0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.112  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::558d:f33d:1ca7:1b9c  prefixlen 64  scopeid 0xfd<compat,link,site,host>
        ether 68:14:01:b4:32:b5  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

cdac@DESKTOP-150G8G6:~$
```

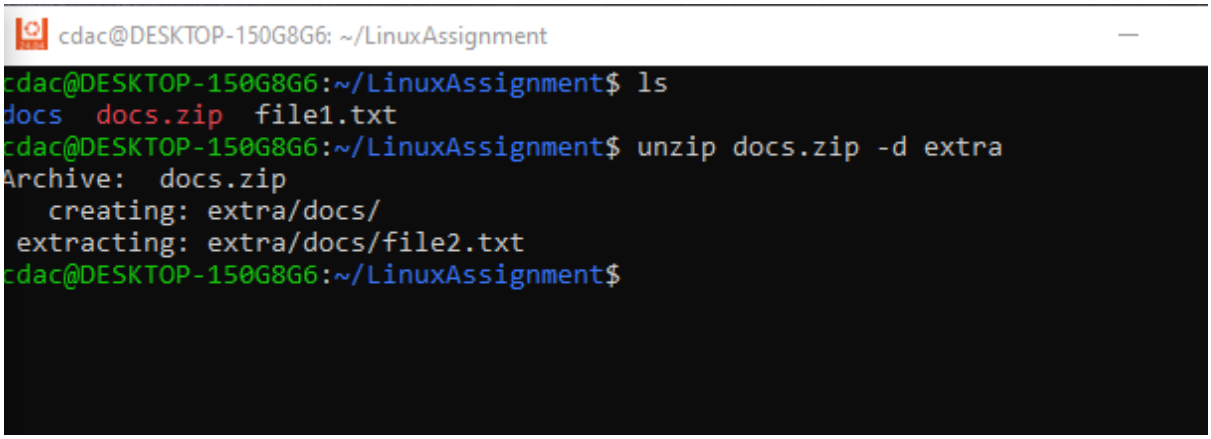
b. Ping a remote server to check connectivity (provide a remote server address to ping).

```
cdac@DESKTOP-150G8G6: ~
cdac@DESKTOP-150G8G6:~$ ping -c 4 134.23.2.3
PING 134.23.2.3 (134.23.2.3) 56(84) bytes of data.

--- 134.23.2.3 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3063ms

cdac@DESKTOP-150G8G6:~$
```

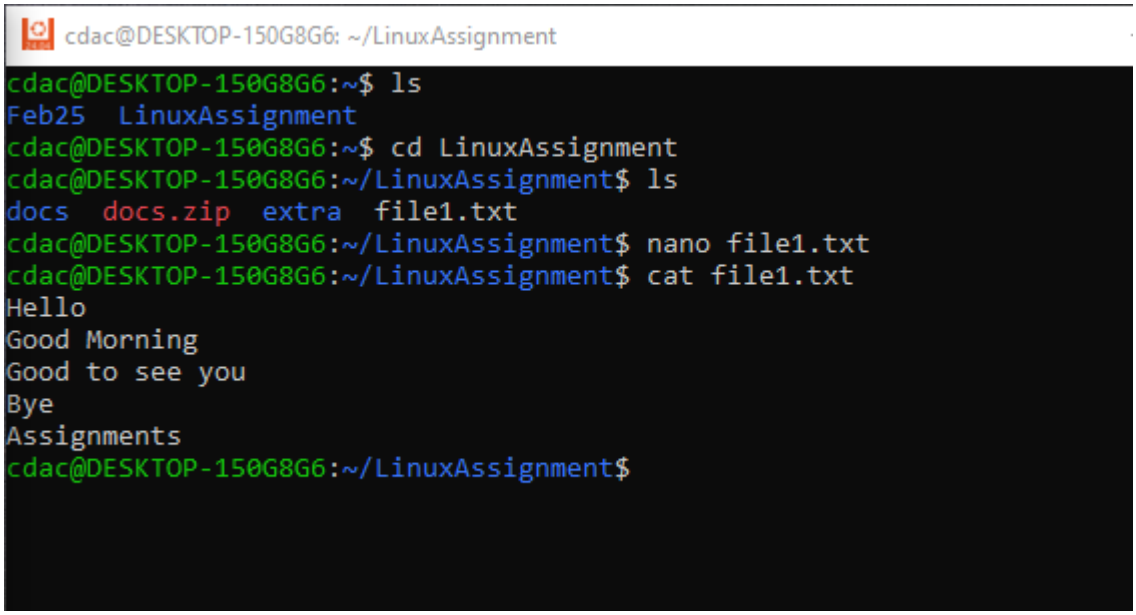
j) File Compression: a. Compress the "docs" directory into a zip file.



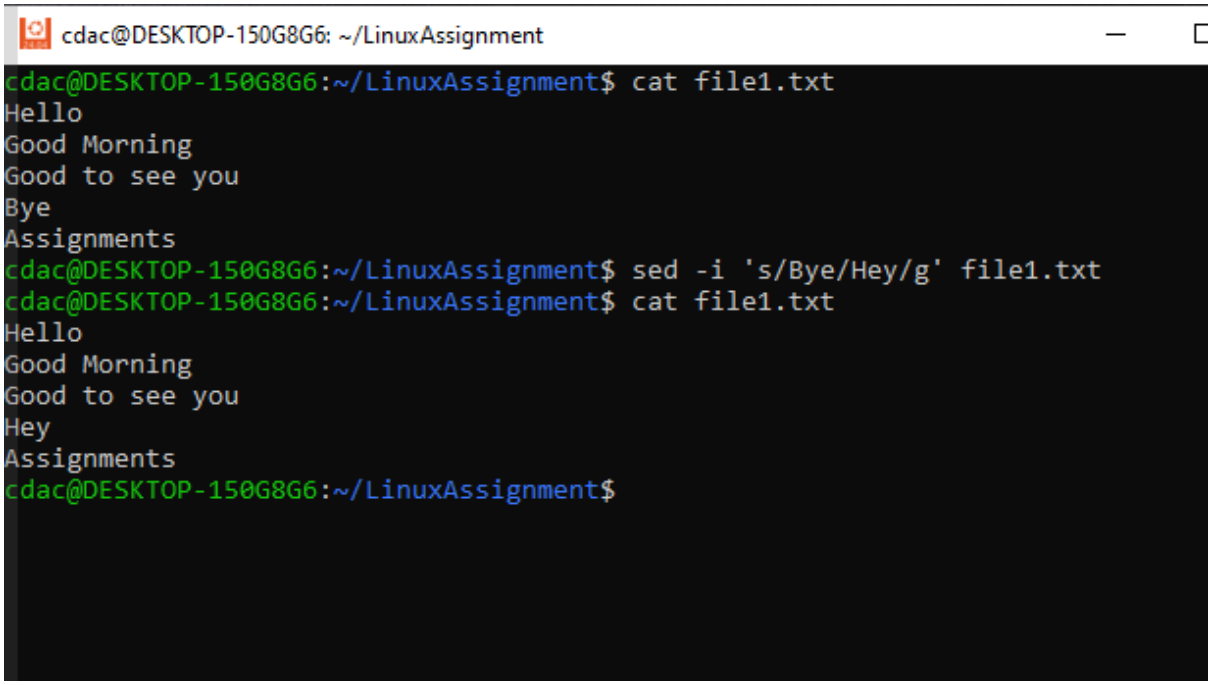
b. Extract the contents of the zip file into a new directory.



k) File Editing: a. Open the "file1.txt" file in a text editor and add some text to it.

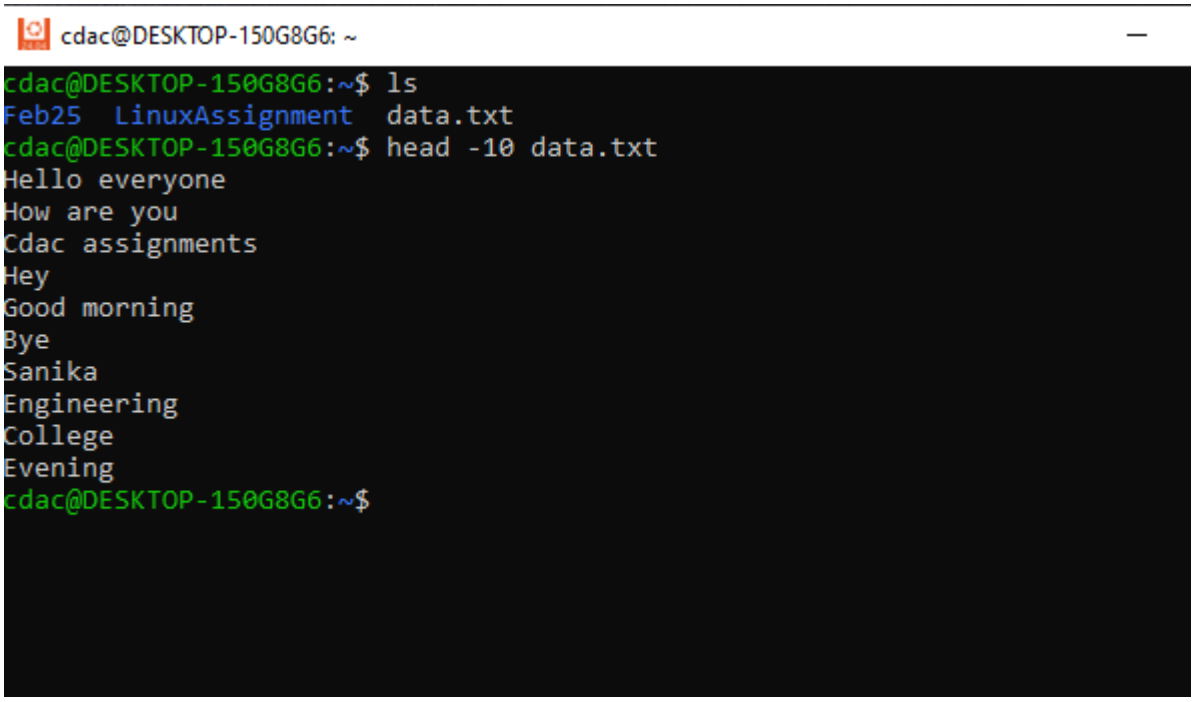


b. Replace a specific word in the "file1.txt" file with another word (provide the original word and the word to replace it with).

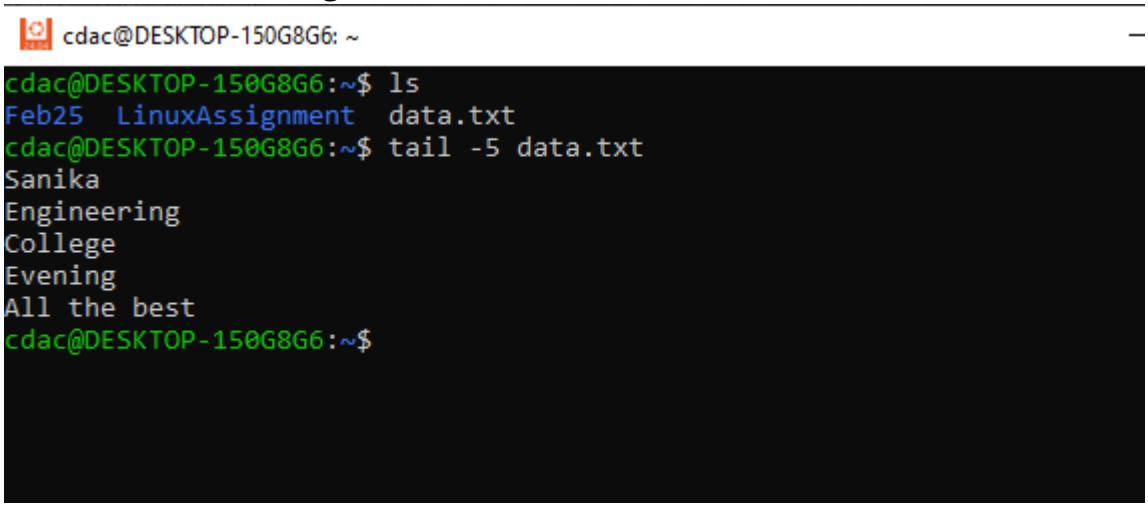


Problem 2

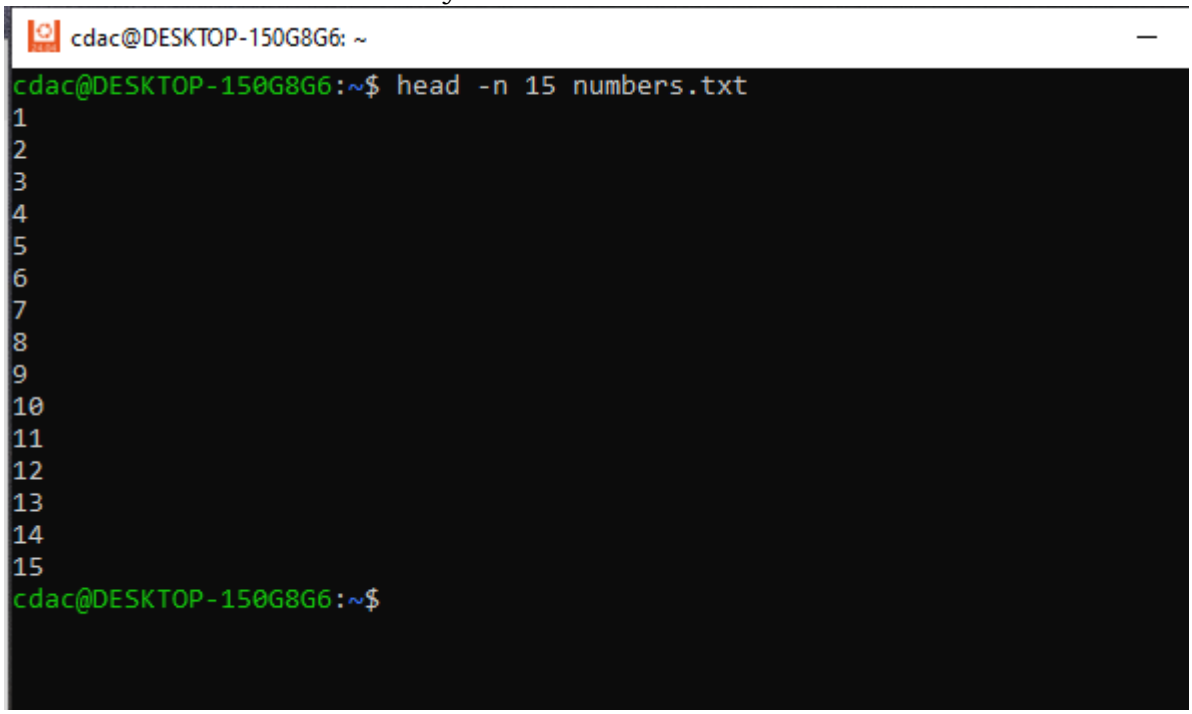
a. Suppose you have a file named "data.txt" containing important information. Display the first 10 lines of this file to quickly glance at its contents using a command.



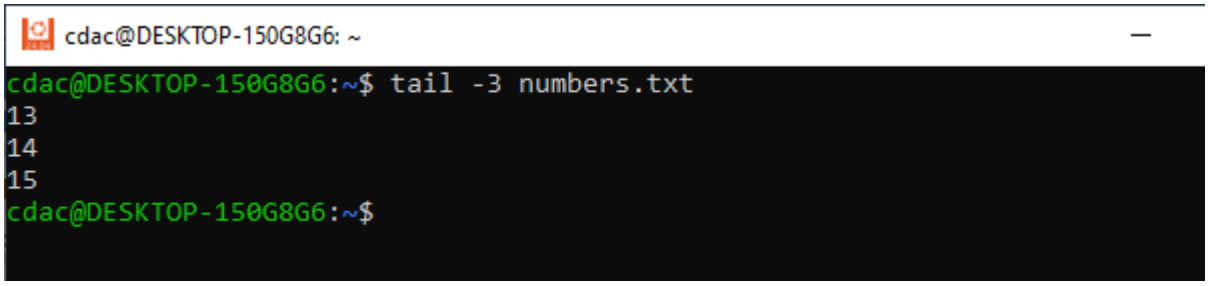
b. Now, to check the end of the file for any recent additions, display the last 5 lines of "data.txt" using another command.



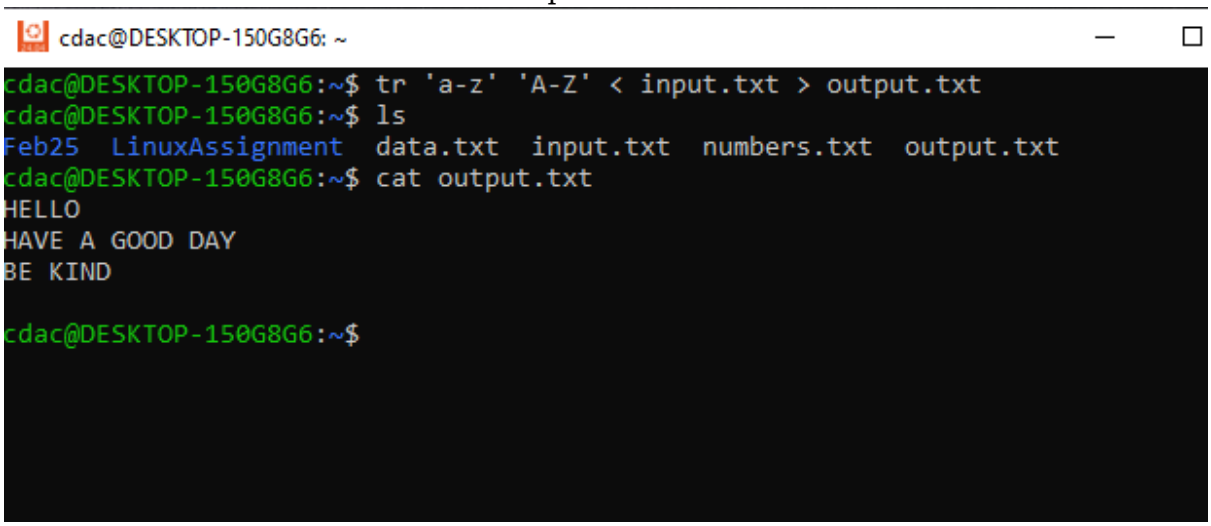
c. In a file named "numbers.txt," there are a series of numbers. Display the first 15 lines of this file to analyze the initial data set.



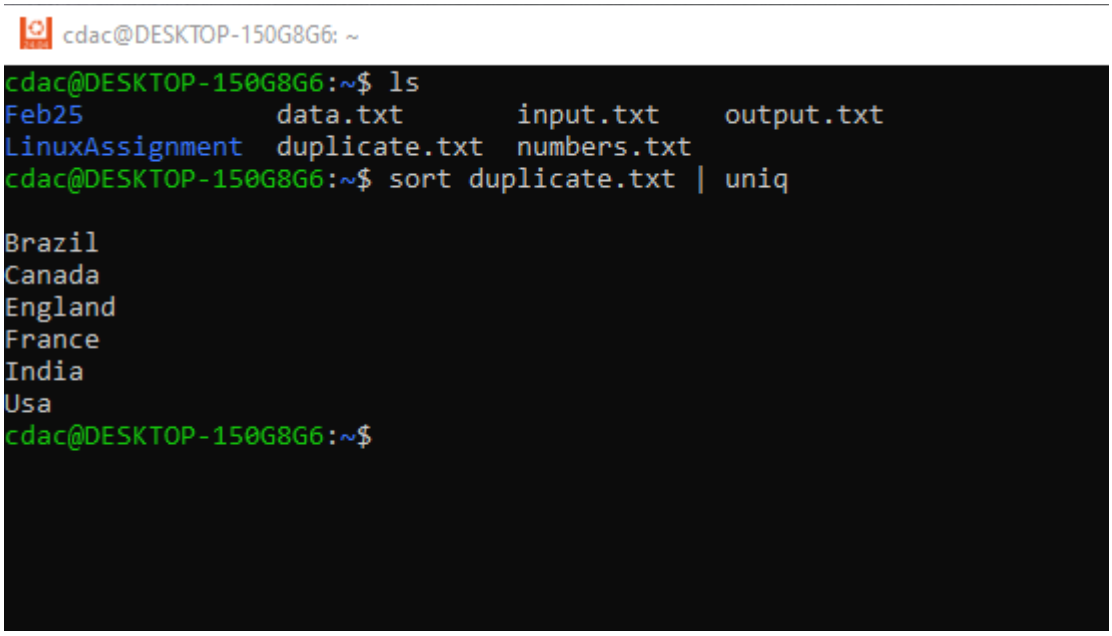
d. To focus on the last few numbers of the dataset, display the last 3 lines of "numbers.txt".



e. Imagine you have a file named "input.txt" with text content. Use a command to translate all lowercase letters to uppercase in "input.txt" and save the modified text in a new file named "output.txt."



f. In a file named "duplicate.txt," there are several lines of text, some of which are duplicates. Use a command to display only the unique lines from "duplicate.txt."



g. In a file named "fruit.txt," there is a list of fruits, but some fruits are repeated. Use a command to display each unique fruit along with the count of its occurrences in "fruit.txt."