Lab3:

The main idea of this assignment is to provide hands on experience on the following topics

Grep

Pipe

redirection

read

command Line arguments

head,tail,tr

1. Create a file “poem.txt” with the following lines

We have not wings, we cannot soar;
But we have feet to scale and climb
By slow degrees, by more and more,
The cloudy summits of our time.

The mighty pyramids of stone
That wedge-like cleave the desert airs,
When nearer seen and better known,
Are but gigantic flights of stairs.

The distant mountains, that uprear
Their solid bastions of the skies,
Are crossed by pathways that appear
As we to higher levels rise.

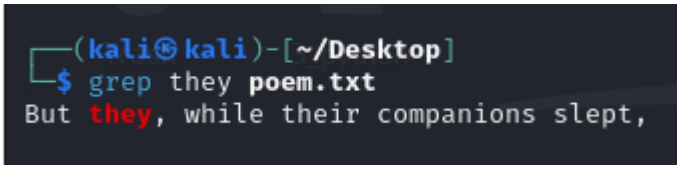
The heights by great men reached and kept
Were not attained by sudden flight,
But they, while their companions slept,
Were toiling upward in the night.

END

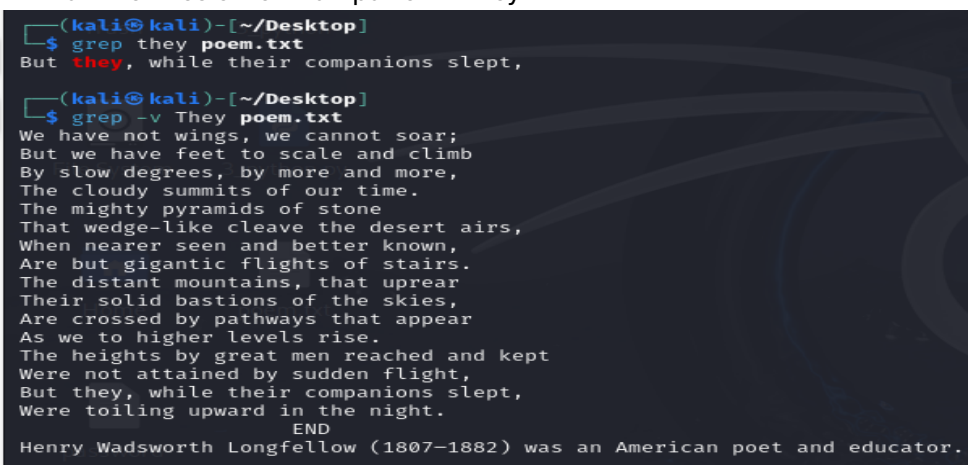
Henry Wadsworth Longfellow (1807–1882) was an American poet and educator.

Do the following task using grep command

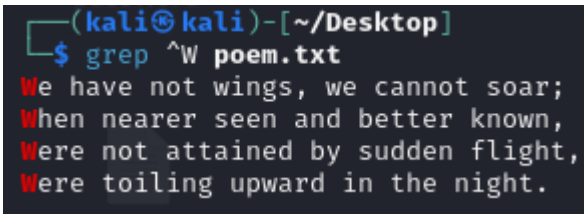
a. Print all the lines with the pattern “they”



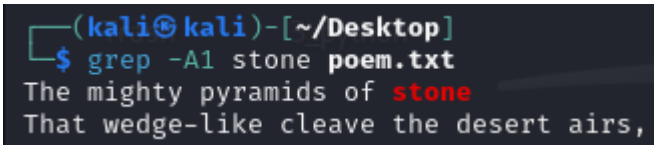
b. Print all the lines other than pattern “They”



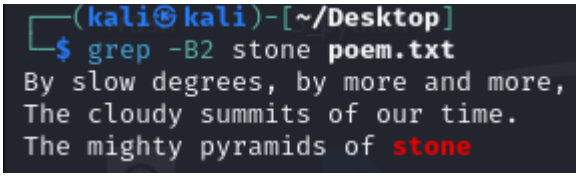
c. Print all the lines starts with “w”



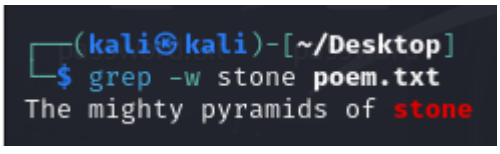
d. Print the next lines after the pattern “stone” matches
Hint: man grep



e. Print the 2 lines above the pattern “stone” matches
Hint: man grep



f. Search the pattern with exact match



2. Explore variations of grep command
   a. ngrep
      A network packet analyzer that uses regular expressions to filter packets. It's useful for monitoring network traffic based on specific patterns.
   b. pgrep
      Searches for processes currently running on the system based on name or other attributes. It returns the process IDs of matching processes.
   c. zgrep
      A variant of `grep` that can search compressed files (gzip). It allows you to search through `.gz` files without needing to decompress them first.

   d. egrep
      An extended version of `grep`, which supports extended regular expressions. It allows for more complex pattern matching without needing to escape certain characters.

3. Write a shell script to get the pattern and filenames from the user and check whether the pattern is present or not.

```
#! /usr/bin/bash

read -p "Enter pattern to search: " pattern
read -p "Enter the filename: " file

if grep -q "$pattern" "$file"; then
    echo " Pattern '$pattern' found in File: '$file'."
else
    echo " Pattern '$pattern' not found in File: '$file'."
fi

```

```
┌──(kali㉿kali)-[~/Desktop]
└─$ ./script.sh
Enter pattern to search: stone
Enter the filename: poem.txt
 Pattern 'stone' found in File: 'poem.txt'.

┌──(kali㉿kali)-[~/Desktop]
└─$ ./script.sh
Enter pattern to search: user
Enter the filename: poem.txt
 Pattern 'user' not found in File: 'poem.txt'.
```

4. Rewrite the above shell script using command line arguments. (pass the pattern and file through command line arguments)

```
#! /usr/bin/bash

if [ $# -ne 2 ]; then
    echo "Usage: $0 pattern filename"
    exit 1
fi

pattern=$1
filename=$2

if grep -q "$pattern" "$filename"; then
    echo "Pattern found."
else
    echo "Pattern not found."
fi
```

```
┌──(kali㉿kali)-[~/Desktop]
└─$ ./script_cli.sh stone poem.txt
Pattern found.

┌──(kali㉿kali)-[~/Desktop]
└─$ ./script_cli.sh user poem.txt
Pattern not found.
```

5. Write a shell script to count total number of regular files in the current working directory.

```
#! /usr/bin/bash

count=$(find . -maxdepth 1 -type f | wc -l)
echo "Total regular files: $count"

```

```
┌──(kali㉿kali)-[~/Desktop]
└─$ ./count_regular.sh
Total regular files: 9
```

6. pipe
   a) Pick the line from 3 to 5.

```
┌──(pb㉿kali)-[~/Desktop]
└─$ sed -n '3,5p' poem.txt
By slow degrees, by more and more,
The cloudy summits of our time.
The mighty pyramids of stone
```

b) List the top 5 largest files in a directory and display their sizes

```
┌──(pb㉿kali)-[~/Desktop]
└─$ ls -lS | head -5
total 116
-rw-rw-r-- 1 pb pb 54295 Nov 12 13:46 MiniPro.zip
drwxrwxr-x 5 pb pb  4096 Aug  8 17:27 CYS
drwxrwxr-x 5 pb pb  4096 Aug 20 18:45 LS
drwxrwxr-x 7 pb pb  4096 Nov 12 13:28 MiniPro
```

c) Print the last 2 modified file details

```
┌──(pb㉿kali)-[~/Desktop]
└─$ ls -lt | head -3
total 116
-rw-rw-r-- 1 pb pb   666 Nov 12 21:13 poem.txt
-rw-rw-r-- 1 pb pb  1836 Nov 12 15:28 yara.txt
```

7. Redirection

a) Convert uppercase into lowercase characters

```
┌──(pb㉿kali)-[~/Desktop]
└─$ tr '[:upper:]' '[:lower:]' < poem.txt > poem_upper.txt
```

b) List the contents of your current directory, including the ownership and permissions, and redirect the output to a file called contents.txt within your home directory.

```
┌──(pb㉿kali)-[~/Desktop]
└─$ ls -l > ~/contents.txt
```

c) Rewrite the shell script (3) using <<

```
┌──(pb㉿kali)-[~/Desktop]
└─$ cat << EOF
heredoc>
heredoc> #! /usr/bin/bash
heredoc>
heredoc> read -p "Enter patttern to search: " pattern
heredoc> read -p "Enter the filename: "file
heredoc>
heredoc> if grep -q "$pattern" "$file"; then
heredoc>         echo "Pattern '$pattern' found in File: '$file'."
heredoc> else
heredoc>         echo "Pattern '$pattern' not found in File: '$file'."
heredoc> fi
heredoc> EOF

#! /usr/bin/bash

read -p "Enter patttern to search: " pattern
read -p "Enter the filename: "file

if grep -q "" ""; then
        echo "Pattern '' found in File: ''."
else
        echo "Pattern '' not found in File: ''."
fi
```

“You don't have to see the whole staircase, just take the first step.”

— **Martin Luther King Jr.**