**Linux\_Lab-5**

#fix the error

/!bin/bash

fruit1 = Apples

fruit2 = Oranges

if [ $1 -lt $# ]

then

echo "This is like comparing $fruit1 and $fruit2!"

elif test [$1 -gt $2 ]

then

echo '$fruit1 win!'

else

echo "Fruit2 win!"

done

=>

#!usr/bin/bash

fruit1='Apples'

fruit2='Oranges'

echo "This is like comparing $fruit1 and $fruit2!"

if [ "$fruit1" \< "$fruit2" ]

then

echo "$fruit1 win!"

else

echo "$fruit2 win!"

fi

More on Grep

1. Print all the lines having the word "pattern".



1. Pick out the blank lines in the file

A computer screen shot of a computer code

Description automatically generated

1. Count total number of empty lines in the file.

A computer screen with text

Description automatically generated

1. Print the line which have both “Sir and Madam”.

A computer screen with text

Description automatically generated

1. pick out lines with “pattern1” “pattern2” or “pattern3”. (use the alternator |)

A computer screen with text

Description automatically generated

1. pick out lines that have at least two p's followed by any number of letters followed by 'ore'. The p's do not have to be next to each other.

A computer screen with text

Description automatically generated

1. pick out all the lines with v, z or I in them

A computer screen with text

Description automatically generated

1. pick out all the lines that do not start with an uppercase letter.

A computer screen shot of a computer code

Description automatically generated

1. pick out all the lines that end with a dash –

A computer screen with white text

Description automatically generated

1. pick out all the words that end with ore

A screen shot of a computer

Description automatically generated

1. pick out all the words that start with f or F

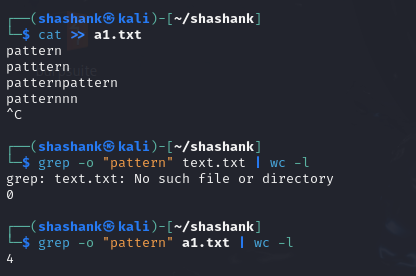
A screenshot of a computer code

Description automatically generated

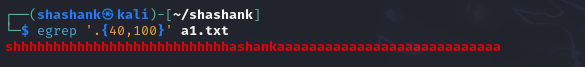
1. pick out lines that uses first letter alliteration - starting two words with the same letter.



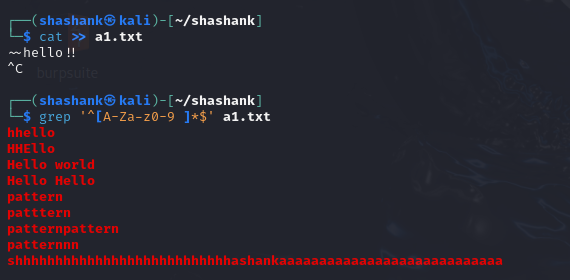
1. determine how many times contains the word "pattern".



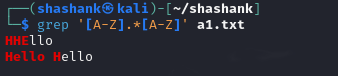
1. to pick out lines with at least 40 characters:



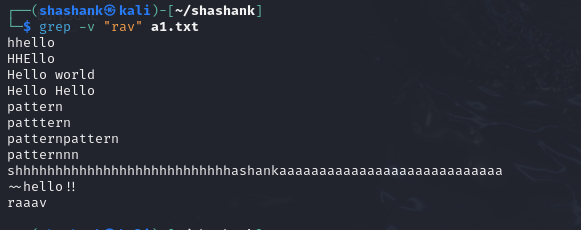
1. to pick out lines with no punctuation



1. to pick out lines with an uppercase letter other than the first character. (The first character on the line does not count.)



1. To pick out lines without rav



Quotes:

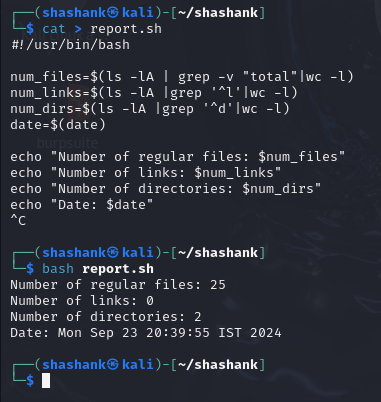
18. Write a shell script to generate a report with the following details.

- Number of regular files

- Number of links

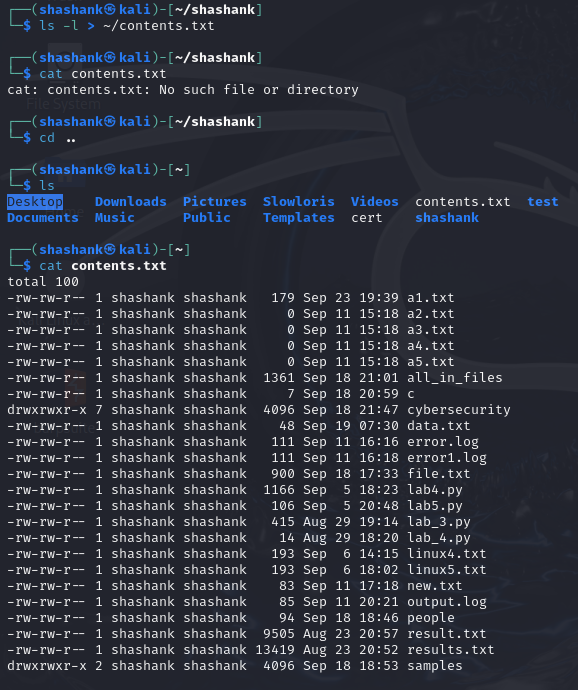
- Number of directories

- Print the date when it was processed!

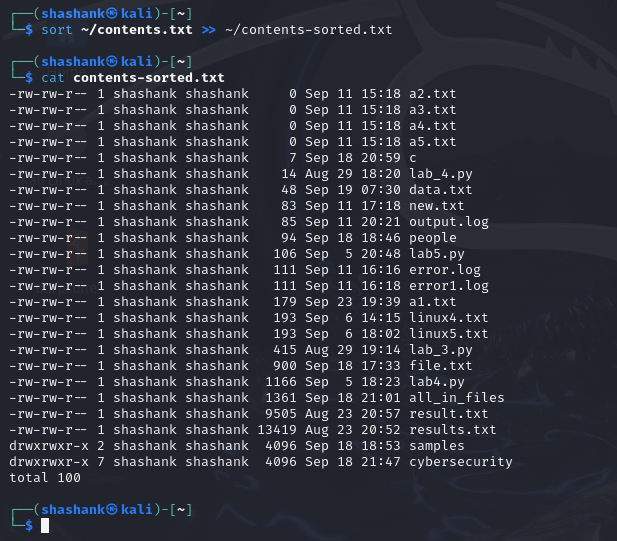


Redirection

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



1. Sort the contents of the contents.txt file from your current directory and append it to the end of a new file named contents-sorted.txt.



1. Display the last 10 lines of the /etc/passwd file and redirect it to a new file in the your user’s Documents directory.

A screenshot of a computer screen

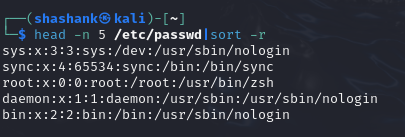
Description automatically generated

1. Count the number of words within the contents.txt file and append the output to the end of a file field2.txt in your home directory. You will need to use both input and output redirection.

A computer screen with white text and blue text

Description automatically generated

1. Display the first 5 lines of the /etc/passwd file and sort the output reverse alphabetically.



1. Using the previously created contents.txt file, count the number of characters of the last 9 lines.

A white text on a black background

Description automatically generated