Experiment no. 6

**Aim**:

Execute the following scripts using grep / sed commands:

1. Write a script using grep command to find the number of words character, words and lines in a file.
2. Write a script using sed command to replace all occurrences of particular word in given a file.

b) Execute the following scripts using awk / perl languages:

1. Write an awk script to print all even numbers in a given range.
2. Write a perl script to check a number is prime or not.

**Outcome:**

LO2: Identify the Unix general purpose commands

LO3: Apply Unix commands for system administrative tasks such as file system management and user management.

LO6: Implement advanced scripts using awk & perl languages and grep, sed, etc. commandsfor performing various tasks.

**Theory:**

## GREP Command

The grep filter searches a file for a particular pattern of characters, and displays all lines that contain that pattern. The pattern that is searched in the file is referred to as the regular expression (grep stands for global search for regular expression and print out). Syntax: grep [options] pattern [files]

## SED Command

SED command in UNIX stands for stream editor and it can perform lots of functions on file like searching, find and replace, insertion or deletion. Though most common use of SED command in UNIX is for substitution or for find and replace. By using SED you can edit files even without opening them, which is much quicker way to find and replace something in file, than first opening that file in VI Editor and then changing it. SED is a powerful text stream editor. Can do insertion, deletion, search and replace(substitution).

SED command in unix supports regular expression which allows it perform complex pattern matching.

Syntax: sed OPTIONS... [SCRIPT] [INPUTFILE...]

## AWK Script

Awk is a scripting language used for manipulating data and generating reports. The awk command programming language requires no compiling and allows the user to use variables, numeric functions, string functions, and logical operators.

Awk is a utility that enables a programmer to write tiny but effective programs in the form of statements that define text patterns that are to be searched for in each line of a document and the action that is to be taken when a match is found within a line. Awk is mostly used for pattern scanning and processing. It searches one or more files to see if they contain lines that matches with the specified patterns and then perform the associated actions.

Awk is abbreviated from the names of the developers – Aho, Weinberger, and Kernighan.

## Perl Script

Perl is a general-purpose, high level interpreted and dynamic programming language. At the beginning level, Perl was developed only for system management and text handling but in later versions, Perl got the ability to handle regular expressions, and network sockets, etc. At present Perl is popular for its ability to handling the Regex(Regular Expressions). The first version of Perl was 1.0 which was released on December 18, 1987. Perl 6 is different from Perl 5 because it is a fully object-oriented reimplementation of Perl 5.

# Write a script using grep command to find the number of words character, words and lines in a file.

echo Enter filename:

read file

if [ ! -f $file ] then

echo -e "File $file doesn't exist" exit echo File exist.

fi

w = 'cat $file | wc -w' c = 'cat $file | wc -c' l

= 'grep -c "."$file' echo -e "\nNumber of characters in $file is $c" echo Number of words in $file is

$w echo Number of lines in $file is $l echo -e "\nThe content in the file:" cat $file echo -e "\n"

# Output:

# A screenshot of a computer Description automatically generated

# Write a script using sed command to replace all occurrences of particular word in given a file.

**Write an awk script to print all even numbers in a given range.**

BEGIN {

print "Even numbers from 0 to 10: "; for (i=0; i<=10; i++)

{

if (i%2 == 0) {

print i;

}

}

}

# Output:

# Write a perl script to check a number is prime or not.

#!/bin/perl print "Enter a number: ";

$n = <>; $c

= 0; if ($n

== 2) {

print "$n is a Prime Number";

}

else { for ($i=2; $i<=$n-1; i++)

{

if ($n % $i == 0)

{

$c = 1; break:

}

}

if ($c == 1)

{

print "$n is not a Prime Number\n";

}

else

{

print "$n is a Prime Number\n";

}

}

# Output:

# 

**Conclusion**:

Successfully understood applied perl and awk scripts in Linux. We also executed grep and sed commands.