A close-up of a logo

Description automatically generated with medium confidence

**PRACTICAL JOURNAL OF  
 Linux Programming**

**BTECH: Third-Year**

| **Name of Student** | **:** | **\_\_\_\_\_SUYASH JETHWANI\_\_\_\_\_.** |
| --- | --- | --- |
| **Branch & Section** | **:** | **\_\_\_\_\_\_\_\_CSE (CS4)\_\_\_\_\_\_\_\_\_.** |
| **Roll Number** | **:** | **\_\_\_\_\_\_\_0827CS191240\_\_\_\_\_\_.** |
| **Year** | **:** | **\_\_\_\_III\_\_\_\_.** |

**Department of Computer Science and Engineering**

**AITR, Indore,**

| **S. No.** | **List of Experiments** | **Page No.** | **Date** | **Remarks** |
| --- | --- | --- | --- | --- |
| **I** | **FILE COMMANDS** | **3-5** |  |  |
| **II** | **DIRECTORY COMMANDS** | **6-7** |  |  |
| **III** | **INPUT OUT PUT REDIRECTION AND PIPE** | **8** |  |  |
| **IV** | **GREP COMMAND** | **9** |  |  |
| **V** | **CHMOD COMMAND** | **10** |  |  |
| **VI** | **WILD CARDS** | **10** |  |  |
| **VII** | **PROCESS COMMANDS** | **11-12** |  |  |
| **VIII** | **SHELL SCRIPT TO PRINT SUM OF TWO NUMBERS** | **13** |  |  |
| **IX** | **SHELL SCRIPT TO PRINT GREATEST AMONG TWO NUMBERS** | **13** |  |  |
| **X** | **SHELL SCRIPT TO COPY FILE FROM SOURCE TO DESTINATION** | **13** |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

### EXPERIMENT 1

##### OBJECTIVE

To study file commands:

* + - * Touch
      * Cat
      * Mv
      * Rm
      * Head
      * Tail
      * Wc
      * Sort
      * Cp
      * Cmp
      * Unique
      * Nl
      * Tr
      * Cut
      * Paste
      * Unname

##### DESCRIPTION / PROCEDURE

**Touch command:**

**Use:** To create multiple blank files

**Syntax:** touch <filename>

**Example:**

touch a1 a2 a3

**Cat Command:**

**Use:** To read content of a file

**Syntax:** cat <filename>

**Example:** cat a1

**Cat Command:**

**Use:** Display the content of multiple files at once using cat command.

**Syntax:** cat <filename1>; cat<filename2>; cat<filename3>

**Example:** cat test1; cat test2; cat test3

**Cat Command:**

**Use:** This command is used to concatenate two files

**Syntax:** cat <filename1>, <filename2>**>**<filename3>

**Example:** cat file1 file2 >file3

**Cat-n Command:**

**Use:** To display line numbers of a file.

**Syntax:** cat -n<filename>

**Example:** cat-n test1

**Cat-e Command:**

**Use:** To read content of a file

**Syntax:** cat -e<filename>

**Example:** cat –e test1

**Rm command**

**Use:** This command is used to remove file.

**Syntax:** rm file\_to\_remove

**Example:** rm test1

**Head command**

**Use:** To display the file content up to specific line number from starting.

**Syntax:** head –number filename

**Example:** head -6 abc.txt

**Tail command**

**Use:** To display the content of file of last specific lines.

**Syntax:** tail -number filename

**Example:** tail -5 abc.txt

**Wc command**

**Use:** To show file content in the form of Lines, Word and Character.

**Syntax:** wc <filename>

**Example:** wc file1

**Sort command**

**Use:** This command is used to sort the content of file..

**Syntax:** sort <filename>

**Example:** sort file1

**Cmp command**

**Use:** This command is used to compare two files.

**Syntax:** cmp filename1 filename2

**Example:** cmp abc.txt xyz.txt

**Unique command**

**Use:** This command used to print the same name only once.

**Syntax:** uniq filename

**Example:** uniq test.txt

**Nl command**

**Use:** This command is used to number the lines. It is used to print the line number of the

content.

**Syntax:** nl filename

**Example:** nl test.txt

**Tr command**

**Use:** This command is used for translating character. This command is used to replace the character with other character..

**Syntax:** tr exp1 exp2 < filename

**Example:** tr ‘.’ ‘/’ <file1 (change 07.09.20 to 07/09/20)

**Cut Command**

**Use:** to remove sections from each line of file.

**Syntax:** cut -c1 -4 filename

Example:

Cut -c1-4 a1

**Paste Command**

**Use:** to joint content of two files (vertically)

**Syntax:** paste filename1 filename2

Example:

Paste a1 a2

**Uname Command**

**Use:** to determine the processor architecture, the system hostname and the version of kernel running on the system.

**Syntax:** uname

Example:

uname

### EXPERIMENT 2

##### OBJECTIVE

To study directory commands:

* Mkdir
* Rmdir
* Rm
* Mv
* Cp
* Ls
* Cd
* Pwd
* History

**Mkdir command**

**Use:** mkdir command is used to create a directory.

**Syntax:** mkdir directory name

**Example:** mkdir dir1

**Rmdir command**

**Use:** rmdir allows you to remove or delete directories but not their contents.

**Syntax:** rmdir [options] directories

**Example:** rmdir dir1

**Rm command**

**Use:** rm allows you to remove files or directories.

**Syntax:** rm files

**Example:** rm abc.txt

**Mv command**

**Use:** This command is used to move one or more files or directories from one place to another.

**Syntax:** mv file\_to\_move destination\_place

**Example:** mv test1 html

**Cp command**

**Use:** This command is used to copy the content of one file or directory to another.

**Syntax:** cp file\_to\_copy new\_file\_name

**Example:** cp abc.txt xyz.txt

**Ls command**

**Use:** This command is used to list the directories.

**Syntax:** ls

**Example:** ls

**Cd command**

**Use:** This command is the key command to move around your file structure.

**Syntax:** cd [name of directory you want to move to]

**Example:** cd abc

**Pwd command**

**Use:** The pwd command in Linux translates to "Print Working Directory" and is used to display the path of the current working directory inside the terminal.

**Syntax:** pwd

**Example:** pwd

**History Command**

**Use:** to view history of all the commands previously executed

**Syntax:** history

Example:

history

### EXPERIMENT 3

##### OBJECTIVE

To study Input Output Redirection and Pipes

* + - * Input redirection(<)
      * Output Redirection(> and >>)
      * Pipe command

**DESCRIPTION/PROCEDURE**

As we have seen, many commands such as **ls** print their output on the display. By using some special notations we can *redirect* the output of many commands to files, devices, and even to the input of other commands.

**Input Redirection:**

**Use:** The **‘<**’symbol is used for input redirection

**Syntax:** command **<** filename

**Example**: sort**<**test1.txt

**Output Redirection:**

**Use:** The **‘>**’symbol is used for output redirection

**Syntax:** command **>** filename

**Example**: ls -al **>** test1

If we do not want a file to be overwritten but want append content to an existing file, then we can use **‘>>**’operator

**Syntax:** command >> filename

**Example**: ls >> test1

**Pipe Command:**

**Use:** The | command is called a pipe. It is used to pipe, or transfer, the standard output from the command on its left into the standard input of the command on its right.

**Syntax:** command\_1|command\_2

**Example**: cat file1.txt|sort

**EXPERIMENT 4**

##### OBJECTIVE

To study GREP commands:

* + - * Pattern
      * -c
      * word
      * -i
      * -n
      * -v

**DESCRIPTION/PROCEDURE**

The grep **(Globally Search For a regular expression & print)** command searches a file or file

For lines that match a provided regular expression

**Pattern Command**

**Use:** To search the word in a file.

**Syntax:** grep pattern filename

Example:

grep hi test

**-c Command**

**Use:** to count the same string and print the number.

**Syntax:** grep -c ‘string’ filename

Example:

Grep -c ‘hello’ test

**word Command**

**Use:** to not print the file in which the word specified by user is not present.

**Syntax:** grep word filename1 filename2 filename3

Example:

grep hello test1 test2 test3

**-i Command**

**Use:** to ignore the case sensitive words & show the content.

**Syntax:** grep -i word filename

Example:

grep-i hello test

**word Command**

**Use:** to show the line number of file content.

**Syntax:** grep -n word filename

Example:

grep -n hello test

**-v Command**

**Use:** to ignore the content which users have searched.

**Syntax:** grep-v word filename

Example:

grep -v hello test

**EXPERIMENT 5**

##### OBJECTIVE

##### To study CHMOD COMMAND

**DESCRIPTION/PROCEDURE**

**Use: CHMOD** (Change mode) to change the access mode of a file.

**Syntax:** chmod [reference][operator][mode] file...

**Example:** chmod u=r assgn1\_client.c

**EXPERIMENT 6**

##### OBJECTIVE

##### To study Wild Cards:

* + - * Star or asterisk(\*)
      * Question Mark (?)
      * Square brackets([])

**DESCRIPTION/PROCEDURE**

Wildcard characters are used to define the pattern for searching or matching text on string data in the bash shell. Another common use of wildcard characters is to create regular expressions

**Asterisk (\*)**

**Use:** Used with shell commands for searching files

**Syntax:** command \*

**Example:** ls t\*(command will search and print all filenames of the current directory that starts with the character‘t’.)

**Question Mark (?)**

**Use:** Used when we want to search the exact number of characters.

**Syntax:** command ?

**Example:** ls ???t.txt **(**will search those files which are four characters long, the last character is ‘t’ and the extension of the file is ‘.txt’.)

**Square brackets ([])**

**Use:** Different ranges of characters or group of characters can be used within square brackets ([]) for searching files based on the range.

**Syntax:** command []

**Example:** ls [a-s0-9]\*.\* **(**will search any file whose name contains any character within ‘**a-s**’ and any digit within ‘**0-9**’ and the file extension can be any character.)

**EXPERIMENT 7**

##### OBJECTIVE

To study Process commands:

* + - * bg
      * fg
      * top
      * ps
      * Kill PID
      * Nice
      * Renice
      * Df
      * Free

**DESCRIPTION/PROCEDURE**

**bg Command**

**Use:** to place foreground jobs in background.

**Syntax:** bg [job\_spec ...]

Example:

bg %1

**fg Command**

**Use:** bring a background process to the foreground.

**Syntax:** fg jobID

Example:

fg %4

**top Command**

**Use:** to show the linux processes.

**Syntax:** top [Options]

Example:

top -s (use to top in secure mode)

**ps Command**

**Use:** to list the currently running processes and their PIDs along with some other information depends on different options.

**Syntax:** ps [Options]

Example:

ps -d (To view all the processes except session leaders)

**Kill Command**

**Use:** Kills a process

**Syntax:** kill

Example:

kill -l

**nice Command**

**Use:** starts a process with a given priority

**Syntax:** nice [OPTION] [COMMAND [ARG]...]

Example:

nice -n13 pico myfile.txt

**renice Command**

**Use:** changes priority of an already running process.

**Syntax:** renice priority [ [-p] pid ... ] [ [-g] pgrp ... ] [ [-u] user ... ]

Example

renice +1 987 -u daemon root -p 32

**df Command**

**Use:** gives free hard disk space on your system.

**Syntax:** df

Example:

df

**free Command**

**Use:** gives free RAM on your system.

**Syntax:** free [ OPTION ]

Example:

free

**EXPERIMENT 8**

**OBJECTIVE:**

SHELL SCRIPT TO PRINT SUM OF TWO NUMBERS

a=10

b=20

sum=$(( $a + $b ))

echo "Sum is: $sum"

**EXPERIMENT 9**

**OBJECTIVE:**

SHELL SCRIPT TO PRINT GREATEST AMONG TWO NUMBERS

echo "Enter Num1"

read num1

echo "Enter Num2"

read num2

**if** [ $num1 -gt $num2 ]

then

echo $num1

**else**

echo $num2

**EXPERIMENT 10**

**OBJECTIVE:**

SHELL SCRIPT TO COPY FILE FROM SOURCE TO DESTINATION

srcpath=$1

dstpath=$2

if [ ! -d "$srcpath" ]; then

echo "Source path: $srcpath doesn't exist"

exit 1

fi

mkdir -p "$dstpath"

cp -r "$srcpath/\*""$dstpath"