**LAB ASSIGNMENT - 2**

**Part 1 - Outputs of the commands**

### **1. pwd (Print Working Directory)**

Displays the absolute path of the current working directory.

**Output**: /home/vboxuser

### **2. cd (Change Directory)**

Changes the current directory to the specified one. If no argument is provided, it moves to the home directory.

**Output**: After cd /var, the current directory becomes /var.

### **3. ls (List Directory Contents)**

Lists files and directories in the current directory.

**Output**:

file1.txt

file2.txt

folder1/

### **4. mkdir (Make Directory)**

Creates a new directory.

**Output**: After mkdir new\_folder, a directory named new\_folder is created.

### **5. rm (Remove)**

Deletes files or directories. Use with caution!

**Output**: After rm file1.txt, the file file1.txt is deleted.

### **6. touch**

Creates an empty file or updates the timestamp of an existing file.

**Output**: After touch new\_file.txt, a file named new\_file.txt is created.

### **7. hostname**

Displays the hostname of the system.

**Output**: user-PC

### **8. cat (Concatenate and Display)**

Displays the contents of a file.

**Output** (content of file.txt):

Hello, world!

### **9. chmod (Change Mode)**

Modifies file or directory permissions.

**Example**: chmod 755 file.txt sets the permissions to read-write-execute for the owner and read-execute for others.

### **10. echo**

Prints text to the terminal. Commonly used to display strings or write to a file.

**Output**: echo "Hello" results in: Hello

### **11. grep (Global Regular Expression Print)**

Searches for a specific pattern in files or outputs.

**Output**: After grep "test" file.txt, lines containing "test" are displayed.

### **12. fgrep (Fixed-String Grep)**

Searches for fixed strings (no regex).

**Output**: Similar to grep, but faster for exact matches.

### **13. mv (Move)**

Moves or renames files/directories.

**Output**: After mv file1.txt new\_file.txt, file1.txt is renamed to new\_file.txt.

### **14. cp (Copy)**

Copies files or directories.

**Output**: After cp file1.txt backup/, file1.txt is copied to the backup directory.

### **15. more**

Displays file contents page by page for long files.

**Output**: It shows the first few lines and waits for user input to scroll.

### **16. less**

Similar to more but allows both forward and backward navigation.

**Output**: Provides better control while viewing large files.

### **17. wc (Word Count)**

Counts lines, words, and characters in a file.

**Output**:

10 50 300

(lines, words, characters)

### **18. awk**

A powerful text-processing tool used for pattern scanning and actions.

**Output**: After awk '{print $1}' file.txt, it prints the first field of each line.

### **19. sed (Stream Editor)**

Performs text transformations or searches and replaces directly in files.

**Output**: After sed 's/test/example/g' file.txt, all instances of "test" are replaced with "example."

### **20. tail**

Displays the last few lines of a file. Useful for logs.

**Output**: tail file.txt shows the last 10 lines by default.

**Part 2 - Answers to the following Questions:**

### 

### **1. How to navigate to a Specific Directory?**

Use the cd command followed by the path of the directory you want to navigate to.

**Command**:

cd /path/to/directory

**Example:**

cd /home/user/Documents

This will move us to the "Documents" directory within the "user" folder.

### **2. How to see detailed information about files and directories using** ls**?**

Use the ls command with the -l option to display detailed information. Adding -a will include hidden files, and combining both (ls -la) shows all files and their details. **Command**:

ls -l

**Example Output:**

-rw-r--r-- 1 user group 4096 Mar 22 10:00 file.txt

This displays permissions, owner, group, size, modification date, and name of files.

### **3. How to create multiple directories in Linux using mkdir command?**

Use mkdir with the -p option to create multiple directories, including parent directories if they don't exist.

**Command**:

mkdir -p /path/to/directory1 /path/to/directory2

**Example:**

mkdir -p project1/{docs,src,bin}

This creates project1 and its subdirectories docs, src, and bin.

### **4. How to remove multiple files at once with rm?**

We can specify multiple filenames with the rm command to delete them all.

**Command**:

rm file1 file2 file3

**Example:**

rm file1.txt file2.txt file3.txt

This deletes all three files.

### **5. Can rm be used to delete directories?**

Yes, rm can delete directories, but you must use the -r option to remove directories and their contents recursively.

**Command**:

rm -r directory\_name

**Example:**

rm -r old\_project

This deletes the old\_project directory and everything inside it.

### **6. How Do You Copy Files and Directories in Linux?**

Use the cp command.

* **Copy a file**:

cp source\_file destination\_path

**Example:**

cp file.txt /home/user/Documents/

* **Copy a directory (with its contents)**:

cp -r source\_directory destination\_path

**Example:**

cp -r folder1 /home/user/Documents/

**7. How to Rename a file in Linux Using `mv` Command?**

The `mv` command can rename a file by specifying the new name.

**Command:**

mv old\_filename new\_filename

**Example:**

mv file.txt renamed\_file.txt

### **8. How to Move Multiple Files in Linux Using mv Command?**

We can list multiple source files and specify the target directory.

**Command**:

mv file1 file2 file3 target\_directory/

**Example:**

mv file1.txt file2.txt /home/user/Documents/

### **9. How to Create Multiple Empty Files by Using Touch Command in Linux?**

We can specify multiple filenames with the touch command.

**Command**:

touch file1 file2 file3

**Example:**

touch file1.txt file2.txt file3.txt

### **10. How to View the Content of Multiple Files in Linux?**

Use the cat command with multiple filenames.

**Command**:

cat file1 file2

**Example:**

cat file1.txt file2.txt

This will concatenate and display the contents of both files.

### **11. How to Create a File and Add Content in Linux Using cat Command?**

Use cat and redirect the content into a file.

**Command**:

cat > filename

**Example:**

cat > file.txt

Then type the content and press CTRL+D to save.

### **12. How to Append the Contents of One File to the End of Another File Using cat Command?**

Use the >> operator.

**Command**:

cat source\_file >> target\_file

**Example:**

cat file1.txt >> file2.txt

### **13. How to Use cat Command if the File Has a Lot of Content and Can’t Fit in the Terminal?**

Pipe the cat command to a pager like less or more.

**Command**:

cat file.txt | less

or

cat file.txt | more

### **14. How to Merge Contents of Multiple Files Using cat Command?**

Provide all file names as arguments.

**Command**:

cat file1 file2 > merged\_file

**Example:**

cat file1.txt file2.txt > merged.txt

### **15. How to Use cat Command to Append to an Existing File?**

Use the >> operator with the file name.

**Command**:

cat >> existing\_file

**Example:**

cat >> file.txt

Then type the new content and press CTRL+D.

### **16. What is chmod 777, chmod 755, and chmod +x or chmod a+x?**

* chmod 777: Full permissions (read, write, execute) for everyone.
* chmod 755: Full permissions for the owner; read and execute for others.
* chmod +x or chmod a+x: Adds execute permission to the file for all users.

**Example:**

chmod 777 file.txt

chmod 755 script.sh

chmod +x program

### **17. How to Find the Number of Lines That Match the Given String/Pattern?**

Use the grep -c command.

**Command**:

grep -c "pattern" file.txt

**Example:**

grep -c "error" log.txt

### **18. How to Display the Files That Contain the Given String/Pattern?**

Use the grep -l command.

**Command**:

grep -l "pattern" \*.txt

**Example:**

grep -l "hello" \*.txt

### **19. How to Show the Line Number of a File with the Line Matched?**

Use the grep -n command.

**Command**:

grep -n "pattern" file.txt

**Example:**

grep -n "error" log.txt

### **20. How to Match the Lines That Start with a String Using grep?**

Use the grep '^pattern' syntax.

**Command**:

grep '^pattern' file.txt

**Example:**

grep '^hello' file.txt

### **21. Can the sort Command Be Used to Sort Files in Descending Order by Default?**

The sort command sorts in ascending order by default, but you can use the -r option to sort in descending order.

**Command**:

sort -r file.txt

### **22. How Can I Sort a File Based on a Specific Column Using the sort Command?**

Use the -k option followed by the column number.

**Command**:

sort -k column\_number file.txt

**Example**:

To sort by the second column:

sort -k 2 file.txt